University of Tartu
Faculty of Science and Technology
Institute of Mathematics and Statistics

Farhad Guliyev

Model Selection and AIC

Actuarial and Financial Engineering

Master's Thesis (30 ECTS)

Supervisor: PhD. Joonas Sova

Tartu 2025

MODEL SELECTION AND AIC

Master thesis

Farhad Guliyev

Abstract

This master thesis investigates the role of Akaike Information Criterion (AIC) in choosing statistical models, combining theoretical foundations with practical simulations to evaluate its effectiveness. The theoretical part of the thesis shows that AIC is based on Kullback-Leibler divergence and cross-entropy, emphasizing its role in minimizing information loss while maintaining a balance between model fit and complexity.

In practical analysis, samples from parametric distributions supported on unit interval are simulated to evaluate the effectiveness of AIC in correctly identifying the true model among various competing alternatives. The results demonstrate that AIC successfully prevents overfitting by penalizing excessive parameters. Moreover, the asymptotic behavior of AIC is also analyzed to see if and how the probability of choosing the correct model converges as the sample size increases. In addition, AIC bias-corrected estimate of the cross-entropy is analyzed and compared with other bias-corrected estimates in order to evaluate its effectiveness in reducing bias. Overall, this thesis demonstrates the importance of AIC in model selection by optimizing the trade-off between model fit and complexity.

CERCS research specialisation: P160 Statistics, operations research, programming, financial and actuarial mathematics.

Key Words: Akaike Information Criterion (AIC), model selection, Kullback-Leibler divergence, cross-entropy, overfitting, bias-correction.

MUDELI VALIK JA AIC

Magistritöö

Farhad Guliyev

Lühikokkuvõte

Käesolev magistritöö uurib Akaike informatsioonikriteeriumi (AIC) rolli statistiliste mudelite valikul, sidudes teoreetilisi aluseid praktiliste simulatsioonidega selle tõhususe hindamiseks. Töö teoreetiline osa näitab, et AIC põhineb Kullback-Leibleri kaugusel ja ristentroopial. Rõhutatakse AIC rolli infokao minimeerimisel, säilitades samal ajal tasakaalu mudeli sobivuse ja keerukuse vahel. Praktilises analüüsis simuleeritakse valimeid ühikintervallil defineeritud parameetrilistest jaotustest, hindamaks AIC tõhusust õige mudeli tuvastamisel erinevate konkureerivate alternatiivide hulgast. Tulemused näitavad, et AIC vähendab edukalt ülesobitumise riski, karistades liigsete parameetrite eest. Lisaks analüüsitakse ka AIC asümptootilist käitumist, et näha, kas ja kuidas õige mudeli valimise tõenäosus koondub valimi suuruse kasvades. Samuti analüüsitakse ja võrreldakse AIC-l põhinevat nihke suhtes korrigeeritud ristentroopia hinnangut teiste hinnangutega, et hinnata selle tõhusust nihke vähendamisel. Kokkuvõttes demonstreerib see magistritöö AIC olulisust mudeli valikul, optimeerides tasakaalu mudeli sobivuse ja keerukuse vahel.

CERCS teaduseriala: P160 Statistika, operatsioonianalüüs, programmeerimine, finants- ja kindlustusmatemaatika..

Märksõnad: Akaike infokriteerium (AIC), mudeli valik, Kullback-Leibleri lahknemine, ristentroopia, ülemäärane sobitamine, nihke parandus .

Contents

Introduction

Statistical modeling serves as an essential instrument in modern data analysis, offering a framework to perceive the connections and relations between variables, make predictions, and formulate meaningful conclusions. It is crucial to choose an approriate statistical model, as an inappropriate selection can result in biased estimates, inaccurate forecasts, or misleading interpretations. The tool that is widely used for model selection is the Akaike Information Criterion (AIC), which balances model fit and complexity to prevent both overfitting and underfitting. Introduced by Hirotugu Akaike in 1973, AIC is based on information theory, specifically the Kullback-Leibler (KL) divergence, which measures the information loss that occurs when a proposed model is utilized to estimate the true data-generating process.

This research delves into the theoretical foundations and practical applications of AIC in choosing statistical models. The first chapter is theoretical with a concise introduction in regard to selection of statistical model and its importance. It is followed by the role of AIC in statistical model choosing, its connection to KL divergence and its importance in minimizing information loss. The third chapter, simulation study, supports this by demonstrating the importance of AIC through simulations and practical analysis.

The connection between theory and practice highlights the value of AIC as an effective method for model selection which is discussed in conclusion part of the thesis. The findings stress the importance of careful model selection, balancing simplicity and accuracy to ensure reliable and comprehensible results in statistical analysis.

1 Statistical Models, Model Selection, and the Akaike Information Criterion (AIC)

1.1 Statistical Model Choosing and Why Is It Important?

A statistical model is a mathematical framework designed to represent real-world processes by incorporating randomness and uncertainty, allowing researchers to analyze data, identify patterns, and make predictions(MyGreatLearning, 2024). For the sake of making predictions about future values, we assess how well the particular data fits a specific model (Rand, 2025). Depending on the goal of using statistical models, they can be classified in different ways. For instance, statistical models can be parametric and non-parametric, linear and non-linear, explanatory and predictive.

Now, it is time to talk about the concept and importance of choosing appropriate statistical model. As it was stated earlier, statistical models are essential tools for analyzing data, making predictions, and drawing inferences in various fields. Selecting an appropriate model is crucial because an incorrect choice can lead to biased estimates, poor predictions, and misleading conclusions.

There are different reasons and factors which influence model selection. The model equation itself might be interesting at times; for instance, in a regression situation, researchers are looking for the variables (covariates) that affect a response variable. To put it another way, they are interested in determining which variables should be included in the model and in what format—as part of an interaction, transformed, linearly, etc. One model is to be chosen as the final model from among the many that result from allowing variables to be included or excluded from the model. This is how model selection is traditionally used, and it frequently focuses just on variable selection(Claeskens, 2016).

Usually, though, what is of importance is not the model equation, but rather a forecast or an estimate that may be made using the model. The researcher then wants to build the model in order to carry out this estimation or prediction as efficiently as possible. This procedure results in a targeted model choosing or focused selection(Claeskens, 2016).

In order to obtain the relevant and significant outcomes from the statistical model, it should fit the data. However, sometimes, we encounter with the situations called underfitting and overfitting. A situation when the models with lower number of parameters are preferred, but higher number of parameters used, called overfitting (Hawkins, 2003). When it comes to underfitting, it occurs when the model is too simple or the number of parameters used are less than the optimal statistical model requires to capture the patterns lying in the base of data(Wilson, 2024).

Together, these definitions underscore the risks of imbalance in model complexity. Both overfitting and underfitting are problematic when selecting the best statistical model for a given dataset. Overfitting arises when a model is excessively complex, fitting not only the underlying patterns but also random noise in the data. This often occurs when a model has too many parameters compared to the available sample size or when researchers engage in excessive data exploration without appropriate adjustments. The repercussions are significant: overfitted models perform well on the training data but poorly on unseen data, potentially leading to misleading findings in scientific and practical applications. To prevent overfitting, strategies such as cross-validation, favoring simpler models, applying regularization techniques like AIC and BIC, and testing models on separate datasets are recommended. The key takeaway is that researchers must strike a balance between model complexity and generalizability to ensure reliable and reproducible outcomes.

Underfitting also negatively affects model performance, often leading to flawed decisions based on the model's outputs. An underfitted model fails to generalize effectively to new data, producing unreliable insights and conclusions. This issue

is particularly detrimental in fields requiring high-precision predictions, such as healthcare, finance, and marketing. Additionally, underfitting can waste valuable time and resources, as efforts invested in model development may not yield useful results(Easily, 2024).

A well-chosen model ensures accurate predictions, meaningful insights, and conclusions that reflect the true structure of the data. Therefore, an optimal model must avoid both underfitting and overfitting. The challenge in model selection lies in finding the right balance. In other words, models that are too simple may miss important patterns (underfitting), while overly complex models may capture noise instead of meaningful trends (overfitting).

Choosing the right model is crucial in the fields of statistics and data science. There are several tools for appropriate model selection, and the most widely used is the information criterion. By offering a methodical approach to model comparison, the information criterion assists analysts and researchers in drawing well-informed conclusions. The quality of various statistical models can be assessed and compared using information criteria, which are statistical techniques. They penalize for complexity and offer a quantifiable way to evaluate how well a model matches the data. These criteria can be used to choose the model that describes the data more accurately and simply, guaranteeing meaningful and reliable findings. There are several types of information criteria, but the most popular are Akaike Information Criterion(AIC) and Bayesian Information Criterion(BIC). They are very helpful in achieving the balance between model simplicity and goodness-of-fit which is essential to prevent overfitting that happens when a model is too complex and captures noise instead of the underlying data structure. In the following section more information about the definition of AIC and its usefulness in model selection will be presented.

1.2 AIC and Its Usefulness in Model Selection

Model selection must be based on a reasonable criterion for which model is "best," and this criterion must be founded on model philosophy and model-based statistical inference, taking into account the fact that the data is finite and "noisy." The criterion for each fitted model must be estimable from the data, and it must be consistent with the general framework of statistical inference. This basically indicates that the model selection is reasonable and operates within a Bayesian framework, a likelihood-theoretic framework, or both. Moreover, the criterion needs to be able to quantify the uncertainty that each model is the target best model by allowing the computation of model weights and being reducible to a number for each fitted model given the data(Burnham and Anderson, 2004).

As it was mentioned earlier, information criterion is the most-widely used tool for model selection. AIC is one of the most popular information criteria method used for statistical model choosing. AIC model selection aims to quantify information loss when the real model's probability distribution (f) is approximated by the evaluated model's probability distribution (g). The difference between the true model and the estimated model is given by the Kullback–Leibler information quantity(Wagenmakers and Farrell, 2004).

It was introduced in 1973 by Hirotugu Akaike and it aims to find the best model without overfitting. The formula of AIC is as follows:

$$\mathrm{AIC} = -2\ln(L) + 2k$$

Thus, AIC quantifies the trade-off between fit and simplicity, guiding the choice of practical models. In here, L refers to the maximum log-likelihood and k refers to the number of parameters. The penalty term of the criterion makes sure that the complexity of the model is taken into consideration by counting the number of estimated parameters, since increasing the number of parameters in a model will

result in a higher maximized log-likelihood. AIC-type information criterion aims to ensure a balance between the model's complexity, as determined by the penalty term, and fit, as determined by the maximal log-likelihood. A good model should be simple and fit nicely(Claeskens, 2016).

The idea of using AIC is to choose the model with the lowest expected information loss. In practice, among the several candidate models, the model with the lowest AIC value is considered the best one since it has the lowest expected information loss. In the following chapter of the thesis, the derivation of the AIC will be presented using Kullback-Leibler divergence and its relation to information theory will be shown.

2 Kullback-Leibler Divergence and Akaike Criterion

In this chapter of the thesis, theoretical information, key concepts and principles such as entropy, Kullback-Leibler divergence and etc., are taken from the book "Elements of Information Theory"(Cover and Thomas, 2006) and the article “A New Look at the Statistical Model Identification”(Akaike, 1974).

Let p and q be two distributions on some finite set $\mathcal{X}$. The Kullback-Leibler divergence of p from q is defined as following:

$$d(p|q) = \sum_{x \in \mathcal{X}} p(x) \log \left(\frac{p(x)}{q(x)} \right).$$

Kullback-Leibler divergence(KL-D) also can be defined assuming that $X \sim p$. Then, the following definition is obtained:

$$d(p|q) = \mathbb{E} \left[\log p(X) - \log q(X) \right].$$

KL-D is the expected log-difference between distributions p and q where the expectation is taken based on distribution p. The idea behind KL-D is that it calculates the average amount of surprise that occurs during the observation of X, if the true distribution of X is considered q when in fact it is p. The unit of KL-D is given by the base of the logarithm. Usually in information theoretic context the base is 2 and the distance is measured in bits. In statistics, it is usually e, and the resulting information unit is *nat*.

Because of the fact that KL-D is not a symmetric measure of distance and it does not satisfy the triangle inequality, logically, it does not define a metric. At first look, it can be considered as the poor characteristic, but, it turns out that it is very genuine for KL-D because of it being a measure of surprise. In order to understand

it better, let us take a look at a small example. Consider, the coin flip example, and the possible outcomes of the each flip can be either head or tail. Let $\mathcal{X} = \{\mathrm{H}, \mathrm{T}\}$ be the the sample space and p be the probability distribution of normal coin(which equals to $\frac{1}{2}$). Now, q is defined as the probability distribution of the coin which has "Heads" on both sides(which equals to 1). In that case, $d(p|q) = \infty$, however, on the other hand $d(q|p) = 1$ (using base 2 for the logarithm). This implies that there is no symmetry but it can be explained in the following way. In fact, the outcome of the coin flip will be surprising, if the one believes that the coin has "Heads" on both sides, when in reality it is a normal coin because the probability of it landing "Tails" is $\frac{1}{2}$. On the other hand, the outcome of the flip will not be as surprising as in the previous case, if the one believes that the coin is normal, when in fact it has "Heads" on the both of the sides. This is because of the fact that there is absolutely equal probability for landing "Heads" and "Tails".

2.1 Relation to Information Theory

In the previous sections of the thesis, it was clarified that KL-D lays in the derivation of AIC. However, both of these definitions are strongly related to the information theory which is fundamental for this thesis. In information theory, the information is taken from the source, then passes the stages of compressing and decompressing before it is received by the receiver. The main question that information theory deals with is to design compressor(encoder) and decompressor(decoder) so that the least amount of bits are used for the sake of describing the data(Duchi, 2004). In other words, the idea behind the information theory is that it deals with the way of finding optimal options for encoding some text.

Let $\mathcal{X}$ be a sample space and p be a probability distribution. Any text from sample space $\mathcal{X}$ can be considered identically independently distributed with the distribution of p and encoder can be defined as the operator that assigns each symbol in $\mathcal{X}$ to some different sets. The encoder is considered as the optimal one when it has

the shortest length for the words or symbols based on the distribution p.

We have acquired general understanding about the information theory and now, let us focus on the entropy and cross-entropy. Entropy is the fundamental concept for information theory. Let X be a random variable defined in $\mathcal{X}$ and p be a distribution. The entropy of the random variable X distributed in p is as follows:

$$H(p) = -\sum_{x \in \mathcal{X}} p(x) \log p(x) = -\mathbb{E}[\log p(X)]. \tag{1}$$

This entropy equation assesses the randomness of X or the average amount of information required for getting the value of X. For instance, if X is constant, then the entropy is equal to 0 because there is not any randomness in X. The cross-entropy is another important notation and it is defined as the measure of the difference between two probability distributions for a some given set. The equation of the cross-entropy is as follows:

$$H(p|q) = -\sum_{x \in \mathcal{X}} p(x) \log q(x) = -\mathbb{E}_p[\log q(X)]. \tag{2}$$

Thus, we can get that $H(p|p) = H(p)$. Since the concepts of entropy and cross-entropy were introduced, we can rewrite the KL-D as the cross-entropy equation. So, Kullback-Leibler divergence can be expressed as

$$d(p|q) = H(p|q) - H(p). \tag{3}$$

Additionally, it can be shown that the only measure of randomness that meets axiomatic assumptions like continuity and additivity under independence is entropy. Likewise, an axiomatic definition of KL-D can be provided. We can, for instance, confirm the additivity for independent random variables for both KL-D and entropy. In order to do that, let us take two independent random variables X and Y

which are p and q distributed respectively. Then, the equation is:

$$\begin{aligned} H(X,Y) &= -\mathbb{E}[\log(p(X)q(Y))] \\ &= -\mathbb{E}[\log p(X)] - \mathbb{E}[\log q(Y)] \\ &= H(X) + H(Y). \end{aligned}$$

Similarly for KL-D, if $X_1 \sim p_1$ and $X_2 \sim p_2$ are independent, and $Y_1 \sim q_1$ and $Y_2 \sim q_2$ are independent. Then, we obtain the following equation:

$$\begin{aligned} d(X_1, X_2|Y_1, Y_2) &= \mathbb{E}[\log(p_1(X_1)p_2(X_2)) - \log(q_1(X_1)q_2(X_2))] \\ &= \mathbb{E}[\log p_1(X_1) - \log q_1(X_1)] + \mathbb{E}[\log p_2(X_2) - \log q_2(X_2)] \\ &= d(X_1|Y_1) + d(X_2|Y_2). \end{aligned}$$

2.2 Model Selection

While performing statistical analysis, researchers or analysts face two types of error: error caused by modeling and error caused by estimation. These two errors can be combined and characterized as the "Overall risk".

$$\text{Overall risk} = \text{Risk of modeling} + \text{Risk of estimation}.$$

Quite a lot of research has been conducted on the minimization of the second error (caused by estimation), however the first error is also important because wrong model adds the bias to the estimation. So, finding methods to measure the goodness of fit of various models using an objective criterion becomes crucial since our goal is to reduce total risk. Akaike in his seminal paper used KL-D to find a practical way to compare the modeling risk of different models applied on the same sample. Now, the sums will be replaced by the integrals and KL-D will be defined

accordingly by the density functions $f(x)$ and $g(x)$ because a continuous space $\mathbb{R}$ is of interest.

$$d(f|g) = \int f(x) \ln\left(\frac{f(x)}{g(x)}\right) dx.$$

Similarly, KL-D can be expressed in terms of the cross-entropy, using the definitions (1), (2), and (3), as follows:

$$d(f|g) = H(f|g) - H(f).$$

Imagine, there is a parameter space Θ that is contained in $\mathbb{R}^d$. The densities $g(\cdot|\theta)$ are contained on $\mathbb{R}$ and there is identically independently distributed sample $X_1, \ldots, X_n$ that is f distributed. Now, the function space $\{g(\cdot|\theta) : \theta \in \Theta\}$ can be considered as the statistical model by which the density f(reality) will be estimated. So, the goal is to find a such value of Θ that makes $g(\cdot|\theta)$ closest to the reality f. To simplify the notation we shall identify the symbol θ as function $g(\cdot|\theta)$ when used as an argument of some function operator. In terms of KL-D the best θ is simply the one that minimizes

$$d(f|\theta) = H(f|\theta) - H(f).$$

According to the equation above, $H(f)$ is constant and the best Θ minimizes the cross-entropy $H(f|\theta)$. Now, suppose that θ^* is the parameter value minimizing the cross-entropy($H(f|\theta^*) = \min_\theta H(f|\theta)$). As a result, this minimal cross-entropy provides an objective indicator of how well our model captures reality. In other words, it indicates the degree to which the best estimate of our model may approximate the reality of f, with $d(f|\theta^*) = 0$ if and only if f is present in the model (more precisely, if $g(x|\theta^*) = f(x)$ for a.e. x). So, the following idea regarding statistical model selection is acquired: if the minimal cross-entropy could be found for different models, then the best model can simply be chosen as the one for which this

value is the smallest.

θ^* needs to be estimated from sample and MLE is the commonly used for it. The equation is the following:

$$\hat{\theta}_n = argmax_\theta \sum_{i=1}^{n} \ln g(X_i|\theta).$$

The goal here is not to minimize the theoretical cross-entropy $H(f|\theta^*)$, but to minimize the random cross-entropy $H(f|\hat{\theta}_n)$. It is important to mention that, a.s. convergence $\hat{\theta}_n \to \theta^*$ holds only when f belongs to our model($g(\cdot|\theta^*)$). Now apply maximized log-likelihood and equation is as follows:

$$l_n = \max_\theta \sum_{i=1}^{n} \ln g(X_i|\theta) = \sum_{i=1}^{n} \ln g(X_i|\hat{\theta}).$$

In the equation above, l_n is the maximum log-likelihood of the sample. For large sample size n, taking into account the assumption $f \approx g(\cdot|\theta^*)$,

$$\begin{aligned} -\frac{1}{n} l_n &\approx -\mathbb{E}[\ln f(X_1)] \\ &= -\int f(x) \ln f(x)\, dx \\ &\approx -\int f(x) \ln g(x|\theta^*)\, dx \\ &= H(f|\theta^*) \\ &\approx H(f|\hat{\theta}_n) \end{aligned}$$

As a result of the equation above, it is clearly seen that an estimate of $H(f|\hat{\theta}_n)$ is the average maximum log-likelihood of the sample. One can consider using $-l_n$ as a measure for a goodness of fit for the model, but this estimate is biased and overly favors the models with more parameters. It can lead to overfitting which is undesirable since overfitted models captures noise. Akaike showed that this bias can be asymptotically approximated by the dimension d of the parameter space,

i.e.

$$\mathbb{E}[l_n - nH(f|\hat{\theta}_n)] \approx d,$$

and hence the famous Akaike Information Criterion formula for model selection is obtained:

$$\text{AIC} = 2d - 2l_n.$$

The coefficient 2 is added here for historical reasons and it does not affect the model selection. AIC tends to penalize the overly complex models and reduces the danger of overfitting. The term $2d$ can be interpreted as the penalty term. Different models can be compared based on AIC value on the same sample and the model with the smallest AIC value indicates a better fit for the model.

3 Simulation Study

3.1 Model Selection Simulations

Consider the parametric distribution $\mathcal{P}_k(p_1, \ldots, p_{k+1})$, where $p_i > 0$, $\sum_{i=1}^{k+1} p_i = 1$, with the corresponding probability density

$$f_k(x|p_1, \ldots, p_{k+1}) = (k+1) \sum_{i=1}^{k+1} \mathbb{I}_{[(i-1)/(k+1), i/(k+1))}(x) p_i,$$

where:

- $\mathbb{I}_A(x)$ is the indicator function for set A (equal to 1 if $x \in A$ and 0 otherwise)
- The number of free parameters (degrees of freedom) for this model is k
- The interval $[0, 1)$ is divided into $k+1$ equal subintervals $[\frac{i-1}{k+1}, \frac{i}{k+1})$ for $i = 1, \ldots, k+1$

In order to have a better understanding of this probability density function, let us visualize a plot and, based on it, explain the idea behind. Figure 1 represents a piecewise probability density function(pdf) defined over the interval $[0, 1)$, which is divided into three equal subintervals: $[0, \frac{1}{3})$, $[\frac{1}{3}, \frac{2}{3})$, and $[\frac{2}{3}, 1)$ The x-axis of the Figure 1 corresponds to these intervals, marking the boundaries at 0.33, 0.67, and 1. The y-axis represents the density height, which is calculated as $(k+1)p_i$, where p_i is associated with each interval. Here, $k = 2$, so the interval $[0, 1)$ is divided into 3 subintervals.

The goal of the Figure 1 is to show how the probability mass is distributed across the intervals. For instance, if the probabilities are $p = (0.5, 0.3, 0.2)$, the density heights would be 1.5, 0.9, and 0.6, respectively. This ensures that the total area under the histogram sums to 1, maintaining the normalization condition required for a valid probability density function:

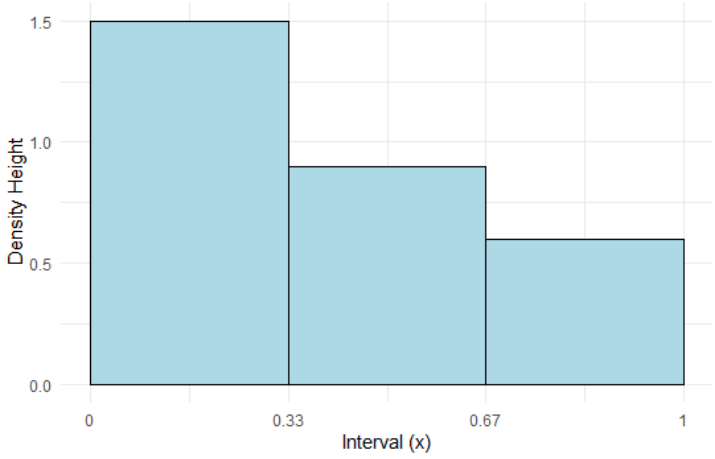

Figure 1: Probability density function visualization

$$\int_0^1 f_k(x)\,dx = \sum_{i=1}^{k+1} \text{height}_i \times \text{width}_i = \sum_{i=1}^{3} (3p_i) \times \frac{1}{3} = \sum_{i=1}^{3} p_i = 1.$$

The uniform distribution, where all probabilities are equal ($p_i = \frac{1}{3}$), would result in equal density heights of 1 across all intervals. The histogram effectively captures the underlying probability structure, with taller bars indicating intervals with higher probability mass.

Now, let us show that the MLE for $(p_1, \ldots, p_{k+1})$ is $\hat{p}^n = (\hat{p}_1, \ldots, \hat{p}_{k+1})$, where

$$\hat{p}_i = \frac{n_i}{n}, \quad n_i = \sum_{j=1}^{n} \mathbb{I}_{[(i-1)/(k+1), i/(k+1))}(x_j).$$

Let us define the likelihood function, and then, by substituting each density function we obtain:

$$\begin{aligned} L(p_1, \ldots, p_{k+1}) &= \prod_{i=1}^{n} f_i(x|p_1, \ldots, p_{k+1}) \\ &= \prod_{j=1}^{n} (k+1) \left(\sum_{i=1}^{k+1} I_{(i-1)(k+1), i/(k+1)}(x) p_i \right). \end{aligned}$$

Now, by taking the logarithm, we obtain log-likelihood function:

$$l(p_1, \ldots, p_{k+1}) = \sum_{j=1}^{n} \ln \left[(k+1) \left(\sum_{i=1}^{k+1} I_{(i-1)(k+1),i/(k+1)}(x) p_i \right) \right].$$

Recall that $n_i = \sum_{j=1}^{n} I_{(i-1)(k+1),i/(k+1)}(x_j)$. Then, the equation above can be rewritten in the following way:

$$\begin{aligned} l(p_1, \ldots, p_{k+1}) &= \sum_{j=1}^{n} \ln(k+1) + \sum_{i=1}^{k+1} n_i \ln p_i \\ &= \sum_{i=1}^{k+1} n_i \ln p_i + n \ln(k+1). \end{aligned}$$

We need to maximize the log likelihood function subject to the constraint $\sum_{i=1}^{k+1} p_i = 1$. For this reason, let us define the Lagrangian function:

$$\mathcal{L}(p_1, \ldots, p_{k+1}, \lambda) = \sum_{i=1}^{k+1} n_i \ln p_i + n \ln(k+1) + \lambda \left(1 - \sum_{i=1}^{k+1} p_i \right),$$

where λ is Lagrange multiplier. Taking the derivative with respect to p_i and setting to 0, we get:

$$\frac{\partial \mathcal{L}}{\partial p_i} = \frac{n_i}{p_i} - \lambda = 0.$$

From this it follows:

$$p_i = \frac{n_i}{\lambda}.$$

Let us substitute $\frac{n_i}{\lambda}$ into $\sum_{i=1}^{k+1} p_i = 1$. As a result of this replacement, the following is obtained:

$$\sum_{i=1}^{k+1} \frac{n_i}{\lambda} = 1, \quad \lambda = \sum_{i=1}^{k+1} n_i = n.$$

Now, we replace the λ in the $p_i = \frac{n_i}{\lambda}$ with n and obtain:

$$\hat{p_i} = \frac{n_i}{n}.$$

Now, here the important question is whether the models $P_1, P_2, ...$ are hierarchical or not. In order to check, recall that two models are hierarchical if one model is the special case of the other model. In other words, models $\{P_k\}$ hierarchical (or nested) if for any $k_1 < k_2$, P_{k_1} is a special case of P_{k_2} obtained by constraining parameters. For example, consider the first model P_1 divided into intervals $[0, 0.5)$ and $[0.5, 1)$ and the second model P_2 divided into intervals $[0, \frac{1}{3})$, $[\frac{1}{3}, \frac{2}{3})$, and $[\frac{2}{3}, 1)$. The models P_1 and P_2 are not hierarchical because it is impossible to restrict or constrain the intervals of P_2 in a way that, it replicates the model P_1. Moreover, the interval $[0, 0.5)$ of model P_1 overlaps with the two intervals $[0, \frac{1}{3})$ and $[\frac{1}{3}, \frac{2}{3})$ of model P_2 at the same time. Let us also consider the example where models are hierarchical. Assume that the model P_1 is divided into intervals $[0, 0.5)$ and $[0.5, 1)$ and the model P_3 into $[0, 0.25)$, $[0.25, 0.5)$, $[0.5, 0.75)$ and $[0.75, 1)$. It can be clearly observed that the interval $[0, 0.5)$ of model P_1 can be derived by combining the two intervals $[0, 0.25)$ and $[0.25, 0.5)$ of model P_3. Similarly, combination of the intervals $[0.5, 0.75)$ and $[0.75, 1)$ gives us the second interval $[0.5, 1)$ of the model P_1. So, P_1 is the special case of P_3 obtained by aggregating the intervals which implies that models are hierarchical.

Even if the models $P_1, P_2, \dots$ are not strictly hierarchical, AIC can be still used as an approximate measure for goodness of fit. Let us fix the "correct" values k^* and $p^* = (p_1^*, p_2^*, \dots, p_{k^*+1}^*)$ and let $X_1, \dots, X_n$ be an i.i.d. sample from $P_{k^*}(p^*)$. For each $k = 1, \dots, m$ (choose $m > k^*$) let $\ell_{n;k}$ be the maximal log-likelihood corresponding to the supposed model P_k, i.e.

$$\ell_{n;k} = \sum_{i=1}^{n} \ln f_k(X_i \mid \hat{p}_{n,k}),$$

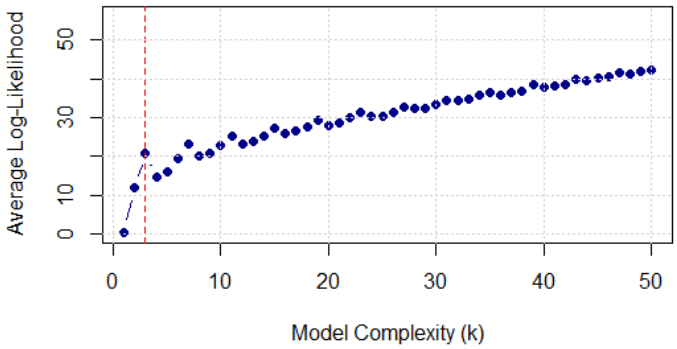

Figure 2: Average Log-Likelihood vs. Model Complexity (k). The plot shows how the log-likelihood changes as the number of free parameters increases.

where $\hat{p}_{n,k}$ is the MLE based on the model P_k. Now, the average values of $l_{n;k}$, AIC and cross-entropy for each $k = 1, ..., m$ will be simulated via R. The goal of this simulation study is to analyze how different model selection criteria behave as the complexity of a probability distribution increases. For these simulations, we assume that we already know the correct number of parameters(k^*) and probabilities(p^*) for the true model. For the illustrated plots, $k^* = 3$ and the probabilities are $p_1^* = 0.4545$, $p_2^* = 0.0455$, $p_3^* = 0.0455$, and $p_4^* = 0.4545$. Random sample data is generated from this model by dividing it into $k^* + 1$ intervals and p^* are the probabilities for each interval. For the simulations, we defined the sample size(n) is 50, maximum number of parameters(m) is 50 and number of simulations(n_{sim}) is 100.

The relationship between the complexity of the model, represented by the number of parameters k, and the average maximum log-likelihood is clearly demonstrated in Figure 2. It is clearly seen from the Figure 2 that as the number of parameters increase, average maximum log-likelihood also goes up. However, this increase does not necessarily indicate that model with more parameters fits data better because model captures noise rather than the underlying patterns. In other words, maximum log-likelihood estimator tends to favor the models with more parameters

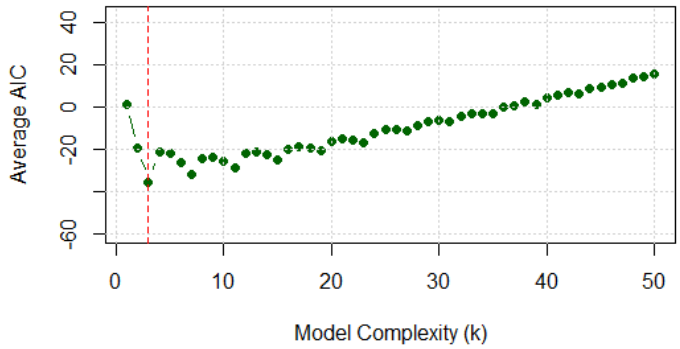

Figure 3: Average AIC vs. Model Complexity (k). The plot shows how the AIC changes as the number of free parameters increases.

which can lead to overfitting. Therefore, we cannot solely rely on the results of maximum log-likelihood as model selection criterion every time.

In search of a better model selection criterion, average AIC measure is plotted and the results are shown in Figure 3. In theoretical part, it was established that the model with the lowest AIC value is considered the better fit than the rest of the models. Earlier, it was also mentioned that the true model appears when $k^* = 3$ and from Figure 3, it is obvious that the lowest AIC value observed at $k = 3$. It indicates that the model with 3 parameters is optimal and the further increase in the number of parameters results in the rise of AIC values, which poses the danger of overfitting. In order to have a confidence regarding the results of AIC plot, we look at the average cross-entropy. For average cross entropy, the following formula is used:

$$H_k^*(\mathbf{X}) = -\int_0^1 f_{k^*}(x|\mathbf{p}^*) \ln f_k(x|\hat{\mathbf{p}}^{n,k}(\mathbf{X}))\, \mathrm{d}x.$$

Based on the given formula, the simulations of average cross-entropy are implemented and the results are given in Figure 4. Earlier, it was identified that the model with the lowest cross-entropy value is considered as the best model since

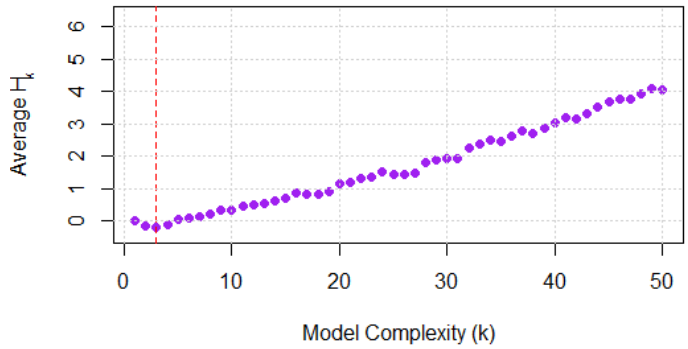

Figure 4: Average Cross-Entropy vs. Model Complexity (k).

it has minimal information loss compared to the rest of candidate models with higher cross-entropy values. According to Figure 4, the lowest cross-entropy value occurs when the number of parameters is equal to 3 and the further increase in the number of parameters lead to rise in average cross-entropy value. Both the results of average AIC and average cross-entropy imply that the best model occurs when $k^* = 3$ which is true considering that earlier the correct model was defined with 3 parameters. So, this observation shows that AIC is a better model selection criterion in comparison to maximum log-likelihood and cross entropy is the cornerstone of AIC.

3.2 Cross-Entropy and Bias Estimates

Let us now analyze the model above theoretically. For theoretical analysis we assume that $\mathcal{P}_k$ is the correct model with correct parameters $\mathbf{p} = (p_1, \ldots, p_{k+1})$. Let $\mathbf{X} = (X_1, \ldots, X_n)$ be an i.i.d. sample from the true distribution $\mathcal{P}_k$. We can express $\ell_{n;k}$ as follows:

$$\begin{aligned}\ell_{n;k} &= \sum_{i=1}^{k+1} n_i(\mathbf{X}) \left(\ln \hat{p}_i(\mathbf{X}) + \ln(k+1)\right) \\ &= n \sum_{i=1}^{k+1} \hat{p}_i(\mathbf{X}) \ln \hat{p}_i(\mathbf{X}) + n \ln(k+1).\end{aligned}$$

Let us also express the cross-entropy H_k^*:

$$\begin{aligned}H_k^*(\mathbf{X}) &= -\int f_k(x|\mathbf{p}) \ln f_k(x|\hat{p}^{n,k}(\mathbf{X})) \, \mathrm{d}x \\ &= -\sum_{i=1}^{k+1} p_i \left(\ln \hat{p}_i(\mathbf{X}) + \ln(k+1)\right) \\ &= -\sum_{i=1}^{k+1} p_i \ln \hat{p}_i(\mathbf{X}) - \ln(k+1). \qquad (1)\end{aligned}$$

Unfortunately, the values p_i are unknown in practice. With the risk of introducing some bias, let us try to estimate these values with $\hat{p}_i(\mathbf{X})$. Substituting those estimates into (1), we obtain

$$H_k^*(\mathbf{X}) \approx -\frac{1}{n} \ell_{n;k}.$$

Unfortunately, this estimate of H_k^* is biased. Let us use Jensen's inequality in order to show that bias is always non-positive.

$$\text{Bias} = -\frac{1}{n} \mathbb{E}[\ell_{n;k}] - \mathbb{E}[H_k^*(\mathbf{X})] \leq 0.$$

From the model, we have:

$$\ell_{n;k} = n \sum_{i=1}^{k+1} \hat{p}_i(\mathbf{X}) \ln \hat{p}_i(\mathbf{X}) + n \ln(k+1),$$

and

$$H_k^*(\mathbf{X}) = -\sum_{i=1}^{k+1} p_i \ln \hat{p}_i(\mathbf{X}) - \ln(k+1).$$

Taking expectations, we get the following:

$$\mathbb{E}[\ell_{n;k}] = n \sum_{i=1}^{k+1} \mathbb{E}[\hat{p}_i(\mathbf{X}) \ln \hat{p}_i(\mathbf{X})] + n \ln(k+1),$$

and

$$\mathbb{E}[H_k^*(\mathbf{X})] = -\sum_{i=1}^{k+1} p_i \mathbb{E}[\ln \hat{p}_i(\mathbf{X})] - \ln(k+1).$$

Substituting these into the bias expression, we obtain the following:

$$\begin{aligned}\text{Bias} = -\frac{1}{n}\left(n \sum_{i=1}^{k+1} \mathbb{E}[\hat{p}_i(\mathbf{X}) \ln \hat{p}_i(\mathbf{X})] + n \ln(k+1)\right) \\ + n \ln(k+1) + \sum_{i=1}^{k+1} p_i \mathbb{E}[\ln \hat{p}_i(\mathbf{X})] + \ln(k+1).\end{aligned}$$

The terms n and $\ln(k+1)$ cancel out, leaving:

$$\text{Bias} = -\sum_{i=1}^{k+1} \mathbb{E}[\hat{p}_i(\mathbf{X}) \ln \hat{p}_i(\mathbf{X})] + \sum_{i=1}^{k+1} p_i \mathbb{E}[\ln \hat{p}_i(\mathbf{X})].$$

The function $\ln(x)$ is concave, so by Jensen's inequality, for any random variable X,

$$\mathbb{E}[\ln(X)] \leq \ln(\mathbb{E}[X]).$$

Substitute X with $\hat{p}_i(\mathbf{X})$:

$$\mathbb{E}[\ln \hat{p}_i(\mathbf{X})] \leq \ln(\mathbb{E}[\hat{p}_i(\mathbf{X})]).$$

We know that $\mathbb{E}[\hat{p}_i(\mathbf{X})] = p_i$. So:

$$\mathbb{E}[\ln \hat{p}_i(\mathbf{X})] \leq \ln(p_i).$$

The function $f(x) = x \ln(x)$ is convex. Therefore, according to Jensen's inequality, for any random variable X,

$$\mathbb{E}[X \ln(X)] \geq \mathbb{E}[X] \ln(\mathbb{E}[X]).$$

Apply this to $\hat{p}_i(\mathbf{X})$:

$$\mathbb{E}[\hat{p}_i(\mathbf{X}) \ln \hat{p}_i(\mathbf{X})] \geq \mathbb{E}[\hat{p}_i(\mathbf{X})] \ln(\mathbb{E}[\hat{p}_i(\mathbf{X})]).$$

Again, since $\mathbb{E}[\hat{p}_i(\mathbf{X})] = p_i$, we have the following:

$$\mathbb{E}[\hat{p}_i(\mathbf{X}) \ln \hat{p}_i(\mathbf{X})] \geq p_i \ln(p_i).$$

Let us recall the Bias estimate formula and substitute the bounds from pervious steps.

$$\text{Bias} = -\sum_{i=1}^{k+1} \mathbb{E}[\hat{p}_i(\mathbf{X}) \ln \hat{p}_i(\mathbf{X})] + \sum_{i=1}^{k+1} p_i \mathbb{E}[\ln \hat{p}_i(\mathbf{X})].$$

Using the bounds:

$$\text{Bias} \leq -\sum_{i=1}^{k+1} p_i \ln(p_i) + \sum_{i=1}^{k+1} p_i \ln(p_i).$$

Simplifying, we get:

$$\text{Bias} \leq 0.$$

Now, let us look at how AIC deals with the bias. AIC tries to correct this bias via

the approximation

$$H_k^*(\mathbf{X}) \approx \frac{k-1}{n}\ell_{n;k}, \tag{2}$$

The goal is to find a better bias-corrected estimate for this specific model. So, we need the following result.

Lemma 1. *For any random variable X the second order Taylor approximation of $\mathbb{E}\ln X$ around $x_0 = \mathbb{E}X$ is*

$$\mathbb{E}\ln X \approx \ln \mathbb{E}X - \frac{\mathbb{V}X}{2(\mathbb{E}X)^2}.$$

Proof. The second-order Taylor expansion of a function $f(X)$ around $x_0 = \mathbb{E}[X]$ is given by:

$$f(X) \approx f(x_0) + f'(x_0)(X - x_0) + \frac{f''(x_0)}{2}(X - x_0)^2.$$

For $f(X) = \ln X$, we compute the following derivatives:

$$f'(X) = \frac{1}{X},$$

and

$$f''(X) = -\frac{1}{X^2}.$$

Let us evaluate these derivatives at $x_0 = \mathbb{E}[X]$:

$$\begin{aligned} f(x_0) &= \ln x_0 = \ln \mathbb{E}[X], \\ f'(x_0) &= \frac{1}{x_0} = \frac{1}{\mathbb{E}[X]}, \\ f''(x_0) &= -\frac{1}{x_0^2} = -\frac{1}{(\mathbb{E}[X])^2}. \end{aligned}$$

Substituting these into the Taylor expansion, we get:

$$\ln X \approx \ln \mathbb{E}[X] + \frac{1}{\mathbb{E}[X]}(X - \mathbb{E}[X]) - \frac{1}{2(\mathbb{E}[X])^2}(X - \mathbb{E}[X])^2.$$

Now, we take the expectation of both sides of the Taylor expansion:

$$\begin{aligned}\mathbb{E}[\ln X] \approx \mathbb{E}\left[\ln \mathbb{E}[X]\right] + \mathbb{E}\left[\frac{1}{\mathbb{E}[X]}(X - \mathbb{E}[X])\right] \\ - \mathbb{E}\left[\frac{1}{2(\mathbb{E}[X])^2}(X - \mathbb{E}[X])^2\right].\end{aligned}$$

After simplifying each term, we get the following:

The first term $\mathbb{E}[\ln \mathbb{E}[X]]$ is a constant, so:

$$\mathbb{E}[\ln \mathbb{E}[X]] = \ln \mathbb{E}[X].$$

The second term $\mathbb{E}\left[\frac{1}{\mathbb{E}[X]}(X - \mathbb{E}[X])\right]$ is:

$$\frac{1}{\mathbb{E}[X]}\mathbb{E}[X - \mathbb{E}[X]] = \frac{1}{\mathbb{E}[X]}(\mathbb{E}[X] - \mathbb{E}[X]) = 0.$$

The third term $\mathbb{E}\left[\frac{1}{2(\mathbb{E}[X])^2}(X - \mathbb{E}[X])^2\right]$ is:

$$\frac{1}{2(\mathbb{E}[X])^2}\mathbb{E}[(X - \mathbb{E}[X])^2] = \frac{1}{2(\mathbb{E}[X])^2}\mathbb{V}[X].$$

Substituting these results back into the equation, we get:

$$\mathbb{E}[\ln X] \approx \ln \mathbb{E}[X] - \frac{\mathbb{V}[X]}{2(\mathbb{E}[X])^2}.$$

□

Since $n\hat{p}_i(\mathbf{X}) \sim Bin(n, p_i)$, we obtain that

$$\mathbb{E}\ln(n\hat{p}_i(\mathbf{X})) \approx \ln(np_i) - \frac{1-p_i}{2np_i},$$

and so

$$\mathbb{E}\ln\hat{p}_i(\mathbf{X}) \approx \ln p_i - \frac{1-p_i}{2np_i}.$$

Let now $H(\mathbf{p})$ denote the base-e entropy of the distribution $\mathbf{p}$, i.e.

$$H(\mathbf{p}) = -\sum_{i=1}^{k+1} p_i \ln p_i.$$

Based on the above we can approximate $\mathbb{E}H_k^*(\mathbf{X})$ as follows:

$$\begin{aligned}\mathbb{E}H_k^*(\mathbf{X}) &\approx H(\mathbf{p}) + \sum_{i=1}^{k+1}\frac{1-p_i}{2n} - (k+1)\ln(k+1)\\ &= H(\mathbf{p}) + \frac{k}{2n} - \ln(k+1).\end{aligned}$$

The entropy $H(\mathbf{p})$ is unknown and needs to be estimated. There are no unbiased estimates for this entropy, but according to Paninski(Paninski, 2003) the commonly used estimates are the naive MLE estimate (also known as empirical entropy).

$$\hat{H}_{MLE} = -\sum_{i=1}^{k+1}\hat{p}_i(\mathbf{X})\ln\hat{p}_i(\mathbf{X}) = -\frac{1}{n}\ell_{n;k} + \ln(k+1), \tag{3}$$

In order to compare and get better bias estimators, Miller-Madow bias corrected entropy estimation and Jacknifed estimation will be used. The information about these estimates is taken from Paninski(Paninski, 2003).

The idea of the Miller-Madow estimator is to correct the bias of $\hat{H}_{MLE}$ by account-

ing for the number of bins(intervals) with non-zero counts. The formula is:

$$\hat{H}_{MM} = \hat{H}_{MLE} + \frac{m-1}{2n}, \tag{4}$$

where:

- m: Number of bins with non-zero counts
- n: Total number of observations
- $\hat{H}_{\text{MLE}}$: Maximum likelihood estimator of entropy

Jackknife bias-corrected estimate is well studied and widely used. The Jackknife estimator reduces bias by computing the entropy multiple times, each time leaving out one observation, and then combining the results. The formula is:

$$\hat{H}_{JK} = n\hat{H}_{MLE} - (n-1)\overline{\hat{H}^{(-j)}_{MLE}}, \tag{5}$$

where:

- $\hat{H}_{MLE}$: MLE of entropy using all n observations
- $\overline{\hat{H}^{(-j)}_{MLE}}$: average of MLE entropy estimates computed by leaving out the j-th observation each time
- n: total number of observations

Both estimators are used to correct or reduce the bias of the maximum likelihood estimate entropy estimation.

Now, we have 4 estimates for $H^*_k(\mathbf{X})$. Simulation study is conducted using these 4 estimates which are naive MLE, AIC-corrected, Miller-Madow and Jackknife estimates. For these simulations, the average values of the correct cross entropy and of each four estimators are estimated via R.

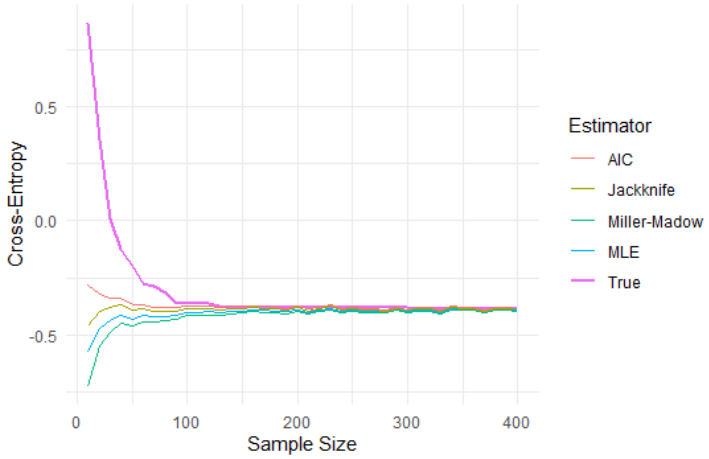

Figure 5: Average Cross Entropy Estimation Comparison

In this simulation study, the aim is to evaluate and compare different estimators of cross-entropy for a probability distribution. The simulation generates data from a probability distribution model by dividing it into k^*+1 intervals where each interval has a specified probabilities p^*. For this analysis, we fixed the model where $k^* = 3$ and the probabilities are $p_1^* = 0.4545$, $p_2^* = 0.0455$, $p_3^* = 0.0455$, and $p_4^* = 0.4545$. This simulation study is important in terms of understanding the behavior of these estimates while converging to the true cross-entropy and bias. Each estimator has distinct bias properties and realizing their behavior helps identify the most reliable method for real-world applications.

For this study, a number of simulations is 100 and as a result, we obtained two figures. As was mentioned earlier, for both figures, the average values of the estimators are plotted. The first shows how the estimators converge towards the true cross-entropy as the sample size increases, while the second illustrates the bias differences by scaling them with the sample size.

The Figure 5 shows how each of 4 bias-corrected estimates behave regarding the true cross-entropy. It is clearly seen that AIC is better but as the sample size(n) increases, all of the 4 estimates converge to true cross-entropy value. It shows the convergence of each of the 4 estimates with the true cross-entropy, but did not

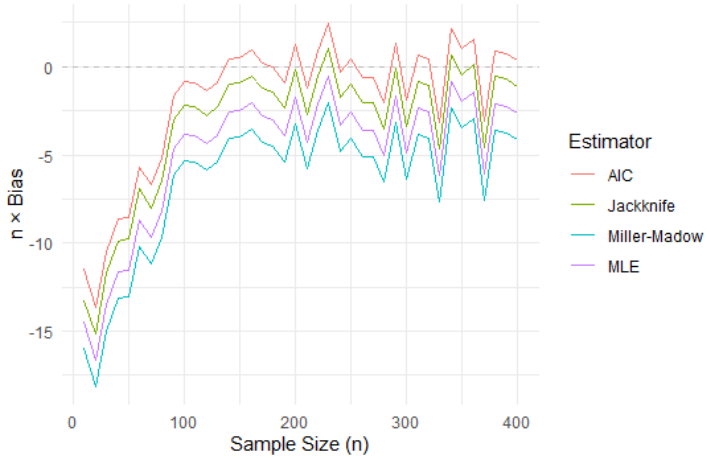

Figure 6: Scaled Bias Comparison: n (Estimate - True Value)

adequately illustrate how biased estimates are. In order to get better understanding of the bias, the Figure 6 was prepared.

According to Figure 6, AIC bias-corrected estimate performs better for small sample sizes, while for larger sample sizes, Jackknifed estimate is better. Miller-Madow estimate and MLE estimate perform worse than the bias-corrected estimates mentioned before. For instance, until $n = 100$, the scaled bias of AIC is the closest to 0, but as sample size increases, it becomes more biased. The possible reason is that for small sample sizes, the penalty term of AIC effectively adjusts for data, whereas for larger sample sizes fixed penalty term becomes less useful. In contrast to AIC bias-corrected estimate, Jackknifed estimate starts poorly for small sample size like $n = 100$, however, as n increases it reduces bias more effectively compared to other estimators. Overall, AIC bias-corrected estimate and Jackknifed estimate performs better in terms of reducing bias than Miller-Madow and MLE estimates. This simulation study showed that for reliable entropy estimation, the choice of method should align with both sample size and the underlying distribution's complexity.

3.3 Asymptotic Behavior of AIC

In this section, the behavior of AIC when the sample size n goes to infinity is investigated. Under general conditions it is true that MLE is a strongly consistent estimator, and therefore if the true model is contained within our list of candidate models, then AIC will select the correct model when $n \to \infty$.

On the other hand if there are multiple correct candidate models (nested inside each other), then ideally we would like to pick the one with the fewest parameters (so that there is not overfitting). However, even though AIC can guard against overfitting in some instances, it does not necessarily guarantee that no overfitting occurs even when $n \to \infty$. Let us take a look at the following example.

Consider the case when there are two candidate models P_0 and P_1. It is assumed that P_0 is the correct model, which is uniform distribution over $[0, 1]$, so that the corresponding density is $f_0(x) = \mathbb{I}_{[0,1]}(x)$. Let P_1 be a distribution with one parameter $p \in [0, 1]$ whose density is

$$f_1(x|p) = 2p\mathbb{I}_{[0,1/2)}(x) + 2(1-p)\mathbb{I}_{[1/2,1]}(x).$$

Then

$$\ell_{n;0} \equiv 0 \quad \text{and} \quad \ell_{n;1} = n[\hat{p}(\mathbf{X}) \ln \hat{p}(\mathbf{X}) + (1 - \hat{p}(\mathbf{X})) \ln(1 - \hat{p}(\mathbf{X}))] + n \ln 2,$$

where $\hat{p}$ is the MLE of p, i.e.

$$\hat{p}(\mathbf{X}) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}_{[0,1/2)}(X_i).$$

Note that P_1 is also a correct model when $p = \frac{1}{2}$ even though it has an extra parameter.

In this case the AIC selects the model P_1 when $1 - \ell_{n;1} < 0$ and model P_0 otherwise.

Equivalently, AIC will choose the model P_1 when

$$n(-h(\hat{p}(\mathbf{X})) - \ln 2) < -1,$$

where $h(\cdot)$ denotes the binary base-e entropy, i.e.

$$h(q) = -q \ln q - (1-q) \ln(1-q).$$

Let now A_n denote the event that the model P_0 is chosen based on AIC (i.e. the event that no overfitting occurs). $P(A_n)$ will be analyzed using simulations via R. Before starting with simulations, we need to implement several computations.

First of all, let us look at AIC selection. The AIC selects $\mathcal{P}_0$ when:

$$\mathrm{AIC}_0 < \mathrm{AIC}_1.$$

For $\mathcal{P}_0$, the AIC is:

$$\mathrm{AIC}_0 = -2\ell_{n;0} + 2 \cdot 0 = 0.$$

For $\mathcal{P}_1$, the AIC is:

$$\mathrm{AIC}_1 = -2\ell_{n;1} + 2 \cdot 1 = -2\ell_{n;1} + 2.$$

Thus, $\mathcal{P}_0$ is selected when:

$$0 < -2\ell_{n;1} + 2,$$
$$\ell_{n;1} < 1$$

Substituting the expression for $\ell_{n;1}$, we get:

$$n\left[\hat{p}(\mathbf{X}) \ln \hat{p}(\mathbf{X}) + (1 - \hat{p}(\mathbf{X})) \ln(1 - \hat{p}(\mathbf{X}))\right] + n \ln 2 < 1.$$

Dividing through by n:

$$\hat{p}(\mathbf{X}) \ln \hat{p}(\mathbf{X}) + (1 - \hat{p}(\mathbf{X})) \ln(1 - \hat{p}(\mathbf{X})) + \ln 2 < \frac{1}{n}.$$

Recall that $h(\hat{p}(\mathbf{X})) = -\hat{p}(\mathbf{X}) \ln \hat{p}(\mathbf{X}) - (1 - \hat{p}(\mathbf{X})) \ln(1 - \hat{p}(\mathbf{X}))$, we rewrite the condition as:

$$-h(\hat{p}(\mathbf{X})) + \ln 2 < \frac{1}{n}.$$

Rearranging:

$$h(\hat{p}(\mathbf{X})) > \ln 2 - \frac{1}{n}.$$

Earlier, we defined binary entropy as follows:

$$h(q) = -q \ln q - (1 - q) \ln(1 - q).$$

It is symmetric around $q = 0.5$, and its restriction to $[0, 0.5]$ is strictly increasing. Therefore, its inverse h^{-1} is well-defined on $[0, \ln 2]$.

The condition $h(\hat{p}(\mathbf{X})) > \ln 2 - \frac{1}{n}$ can be rewritten using h^{-1}. So, we get the following:

$$\hat{p}(\mathbf{X}) > h^{-1}\left(\ln 2 - \frac{1}{n}\right).$$

However, because $h(q)$ is symmetric around $q = 0.5$, the condition $h(\hat{p}(\mathbf{X})) > \ln 2 - \frac{1}{n}$ also implies:

$$\hat{p}(\mathbf{X}) < 1 - h^{-1}\left(\ln 2 - \frac{1}{n}\right).$$

Therefore, the event A_n occurs when:

$$h^{-1}\left(\ln 2 - \frac{1}{n}\right) < \hat{p}(\mathbf{X}) < 1 - h^{-1}\left(\ln 2 - \frac{1}{n}\right).$$

For the simulations, the lower bound, $h^{-1}\left(\ln 2 - \frac{1}{n}\right)$ will be used.

It was previously stated that $n\hat{p}_i(\mathbf{X}) \sim Bin(n, p_i)$. So, let $F(k; n, p)$ denote the distribution function of the binomial distribution $\text{Bin}(n, p)$, i.e.

$$F(k; n, p) = \sum_{i=0}^{\lfloor k \rfloor} \binom{n}{i} p^i (1-p)^{n-i}.$$

Then,

$$P(A_n) = 1 - 2F\big(nh^{-1}(\ln 2 - 1/n); n, 1/2\big).$$

The results of the aforementioned processes are illustrated in Figure 7. For this simulation study, the sample size n is taken as 10000. From the plot, it is observed that as n increases and goes to infinity, $P(A_n)$ converges to a value close to 0.84. From the result of the plot it seems that $P(A_n)$ does not converge to 1. It implies that AIC does not choose the "true" model as $n \to \infty$ and because of that it does not converge to 1. However, AIC shows high predictive accuracy since it converges to 0.84 which is not exactly 1, but close to it.

Let now $t \in (0, 1)$ and consider a more general form for P_1 with density

$$f_1(x|p) = \frac{1}{t} p \mathbb{I}_{[0,t)}(x) + \frac{1}{1-t}(1-p)\mathbb{I}_{[t,1]}(x).$$

In the previous case, $t = \frac{1}{2}$. Let us consider the probability $P(A_n|t)$ and note that t is not a parameter in our model, so, doesn't need to be estimated.

The computations for this simulation study is similar to the previous one. Therefore, we start by estimating the maximum likelihood of the function.

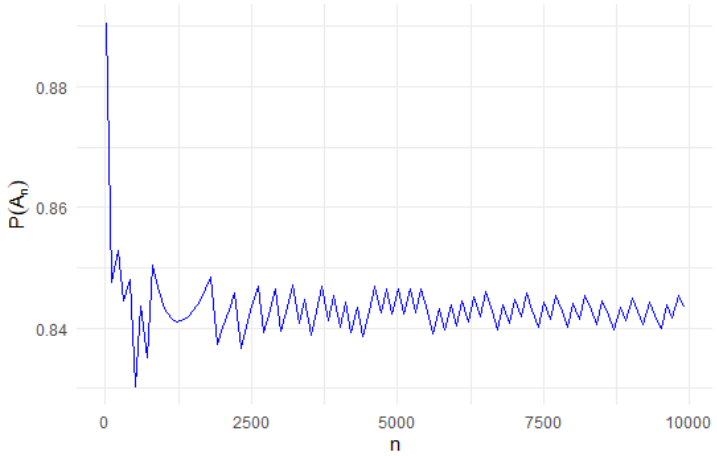

Figure 7: $P(A_n)$ as n icnreases

For a sample $X_1, \ldots, X_n$, the log-likelihood is:

$$\begin{aligned}\ell(p,t) &= \sum_{i=1}^{n} \log f_1(X_i|p,t) \\ &= N_t \ln\left(\frac{p}{t}\right) + (n - N_t)\ln\left(\frac{1-p}{1-t}\right),\end{aligned}$$

where $N_t = \sum_{i=1}^{n} \mathbb{I}_{[0,t)}(X_i)$.

After differentiating, we obtain:

$$\frac{\partial \ell}{\partial p} = \frac{N_t}{p} - \frac{n - N_t}{1-p} = 0.$$

Now after solving for p, it is obtained:

$$\hat{p} = \frac{N_t}{n}.$$

Substituting $\hat{p}$ into the log-likelihood function, we get:

$$\ell_{n;1} = N_t \ln\left(\frac{N_t}{nt}\right) + (n - N_t)\ln\left(\frac{n - N_t}{n(1-t)}\right)$$

After rewriting, we have the following result:

$$\begin{aligned}\ell_{n;1} = n\hat{p}(\mathbf{X}) \ln \hat{p}(\mathbf{X}) + n(1 - \hat{p}(\mathbf{X})) \ln(1 - \hat{p}(\mathbf{X})) \\ - n\hat{p}(\mathbf{X}) \ln t - n(1 - \hat{p}(\mathbf{X})) \ln(1 - t)\end{aligned}$$

Now, we have the maximum log-likelihood function and next we will look to asymptotic behavior.

Like in the previous case, in here we are interested in AIC choosing the model $\mathcal{P}_0$. AIC select the model $\mathcal{P}_0$, when:

$$\text{AIC}_0 < \text{AIC}_1.$$

For $\mathcal{P}_0$, the AIC is:

$$\text{AIC}_0 = -2\ell_{n;0} + 2 \cdot 0 = 0.$$

For $\mathcal{P}_1$, the AIC is:

$$\text{AIC}_1 = -2\ell_{n;1} + 2 \cdot 1 = -2\ell_{n;1} + 2.$$

Thus, $\mathcal{P}_0$ is selected when:

$$\begin{aligned}0 < -2\ell_{n;1} + 2, \\ \ell_{n;1} < 1\end{aligned}$$

Substituting the expression for $\ell_{n;1}$, we get:

$$\begin{aligned}n\hat{p}(\mathbf{X}) \ln \hat{p}(\mathbf{X}) + n(1 - \hat{p}(\mathbf{X})) \ln(1 - \hat{p}(\mathbf{X})) \\ - n\hat{p}(\mathbf{X}) \ln t \\ - n(1 - \hat{p}(\mathbf{X})) \ln(1 - t) < 1.\end{aligned}$$

Dividing by n gives us :

$$\begin{aligned}\hat{p}(\mathbf{X}) \ln \hat{p}(\mathbf{X}) + (1 - \hat{p}(\mathbf{X})) \ln(1 - \hat{p}(\mathbf{X})) \\ - \hat{p}(\mathbf{X}) \ln t \\ - (1 - \hat{p}(\mathbf{X})) \ln(1 - t) < \frac{1}{n}.\end{aligned}$$

The correct model is selected when:

$$\hat{p}(\mathbf{X}) \ln \left(\frac{\hat{p}(\mathbf{X})}{t} \right) + (1 - \hat{p}(\mathbf{X})) \ln \left(\frac{1 - \hat{p}(\mathbf{X})}{1 - t} \right) < \frac{1}{n}.$$

Therefore, consider the function:

$$f(p|t) = p \ln \left(\frac{p}{t} \right) + (1 - p) \ln \left(\frac{1 - p}{1 - t} \right),$$

so that:

$$P(A_n|t) = P(f(\hat{p}(\mathbf{X})|t) < 1/n).$$

The function f is non-negative for $p, t \in (0, 1)$. The minimum of $p \mapsto f(p|t)$ is 0, achieved when $p = t$. The function f is not symmetric about t, so by dividing it into two parts, we get:

$$\begin{aligned} f_1(p) &= f(p|t), \quad p \in (0, t] \\ f_2(p) &= f(p|t), \quad p \in (t, 1) \end{aligned}$$

Now, we define the inverse functions f_1^{-1} and f_2^{-1}. Since $n\hat{p}(\mathbf{X}) \sim \text{Bin}(n, t)$, the probability $P(A_n|t)$ can be expressed through the binomial CDF as:

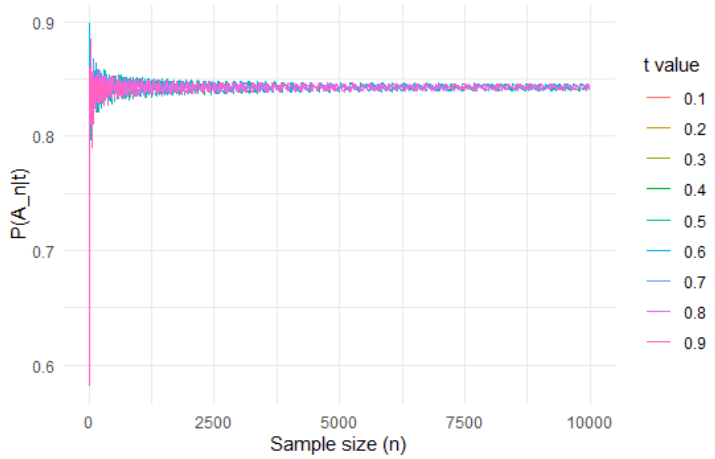

Figure 8: $P(A_n|t)$ as n increases for different values of t

$$P(A_n|t) = P(f_1^{-1}(1/n) \leq \hat{p}(\mathbf{X}) \leq f_2^{-1}(1/n))$$
$$= F(nf_2^{-1}(1/n); n, t) - F(nf_1^{-1}(1/n); n, t).$$

The result of the processes described above is shown in Figure 8. According to Figure 8, initially for smaller values of t, the probability started from above 0.86 and then decreased to the limiting value(close to 0.845) as $n \to \infty$. In contrast, for larger values of t, close to 1, $P(A_n|t)$ started from 0.81 and then reached the limiting value as $n \to \infty$. Overall, for different values of t, as the sample size (n) increases, they all converge to the same value close to 0.845.

Similar to the result of Figure 7, the $P(A_n|t)$ does not converge to 1. In other words, AIC tends to choose the model with the best predictive accuracy rather than "true" model when $n \to \infty$. The reason why both $P(A_n)$ $P(A_n|t)$ do not converge to 1 maybe because of the penalty term of AIC$(2k)$. Since the penalty term is fixed and independent of n, the likelihood term dominates. In its turn, it leads to overfitting which is undesirable for a "true" model. Therefore, $P(A_n)$ and $P(A_n|t)$ do not converge to 1 but close to it, indicating high predictive accuracy.

Conclusion

This thesis investigates the Akaike Information Criterion (AIC) as a methodology for selecting statistical models, integrating theoretical principles with practical simulations to assess its effectiveness. The theoretical study illustrates that AIC, which is based on Kullback-Leibler (KL) divergence, balances between model accuracy and complexity by penalizing excessive parameters to reduce overfitting. The simulations corroborated the effectiveness of AIC in identifying the accurate model among various contenders. The results validated that AIC successfully maintains a balance between goodness of fit and simplicity, with the minimal AIC value corresponding to the true model (for example, $k^{=3}$ in the simulations).

The further analyses gave several important insights about the behavior of AIC. The asymptotic analysis demonstrates that the probabilities $P(A_n)$ and $P(A_n|t)$ converge to a value between 0.84 and 0.845. It indicates that, AIC has a high predictive accuracy but as sample the size increases and goes to infinity, it does not choose the "true" model every time. One possible reason for that can be fixed penalty term($2k$) which effectively prevents overfitting for small sample sizes but for larger ones, may not be that effective. A comparative evaluation of entropy estimators revealed that AIC reduces bias effectively compared to other estimators when the sample size is small, however, as the sample size increases, Jackknife estimator performs better than AIC.

In further research, it can be investigated what is actual theoretical value of the asymptote and why for all different t-values probability converges to the same limit as sample size increases.

References (with BibLaTeX)

Akaike, Hirotugu (1974). "A New Look at the Statistical Model Identification". In: *IEEE Transactions on Automatic Control* 19.6, pp. 716–723. DOI: 10.1109/TAC.1974.1100705. URL: https://ieeexplore.ieee.org/document/1100705 (visited on 05/19/2025).

Burnham, Kenneth P. and David R. Anderson (2004). "Understanding AIC and BIC in Model Selection". In: *Sociological Methods & Research* 33, pp. 261–304. URL: https://journals.sagepub.com/doi/pdf/10.1177/0049124104268644 (visited on 05/19/2025).

Claeskens, Gerda (2016). "Statistical Model Selection". In: *Annual Reviews of Statistics and Its Application*. URL: https://www.annualreviews.org/content/journals/10.1146/annurev-statistics-041715-033413 (visited on 05/19/2025).

Cover, Thomas M. and Joy A. Thomas (2006). *Elements of Information Theory*. Wiley-Interscience.

Duchi, John (2004). *Statistics and Information Theory*. URL: https://web.stanford.edu/class/stats311/lecture-notes.pdf (visited on 05/19/2025).

Easily, Statistics (2024). *What Is Underfitting?* URL: https://statisticseasily.com/glossario/what-is-underfitting/ (visited on 05/19/2025).

Hawkins, Douglas M. (2003). "The Problem of Overfitting". In: *Journal of Chemical Information and Computer Sciences* 43, pp. 1–12. URL: https://pubs.acs.org/doi/epdf/10.1021/ci0342472 (visited on 05/19/2025).

MyGreatLearning (2024). *What is Statistical Modeling?* URL: https://www.mygreatlearning.com/blog/what-is-statistical-modeling/ (visited on 05/19/2025).

Paninski, Liam (2003). "Estimation of Entropy and Mutual Information". In: *Neural Computation* 15, pp. 1191–1253. URL: https://www.cns.nyu.edu/pub/lcv/paninski-infoEst-2003.pdf (visited on 05/19/2025).

Rand, M. (2025). *What is a Statistical Model? – Computational Analysis for Bioscientists*. URL: https://3mmarand.github.io/comp4biosci/what_statistical_model.html (visited on 05/19/2025).

Wagenmakers, Eric-Jan and Simon Farrell (2004). "AIC Model Selection Using Akaike Weights". In: *Psychonomic Bulletin & Review* 11, pp. 192–196. URL: https://link.springer.com/article/10.3758/bf03206482 (visited on 05/19/2025).

Wilson, Kai (2024). *What Is Underfitting and Why You Need to Avoid It?* URL: https://medium.com/@wilsonkai/what-is-underfitting-and-why-you-need-to-avoid-it-ce842bac07b4 (visited on 05/19/2025).

Appendix 1

Code Listings

Code 1: Histogram

```
library(ggplot2)

# Parameters
k <- 2
p <- c(0.5, 0.3, 0.2)  # Probabilities
breaks <- seq(0, 1, length.out = k + 2)  # [0, 1/3, 2/3, 1]
heights <- (k + 1) * p  # [3*0.5=1.5, 3*0.3=0.9, 3*0.2=0.6]

# Data frame
df <- data.frame(
  x_start = breaks[1:(k + 1)],
  x_end = breaks[2:(k + 2)],
  height = heights,
  interval = paste0("[", round(breaks[1:(k + 1)], 2), ",␣",
  round(breaks[2:(k + 2)], 2), ")")
)

# Histogram
ggplot(df) +
  geom_rect(aes(xmin = x_start, xmax = x_end, ymin = 0, ymax = height),
            fill = "lightblue", color = "black", linewidth = 0.5) +
  scale_x_continuous(
    breaks = breaks,
```

```
    labels = paste0(round(breaks, 2))
  ) +
  labs(
    x = "Interval (x)",
    y = "Density Height"
  ) +
  theme_minimal()
```

Code 2: MLE, AIC and Cross Entropy Simulations

```
# True parameters
u <- c(10, 1, 1, 10)
k_star <- length(u) - 1
p_star <- u / sum(u)
p_star



# Simulation function generated using true parameters
simulate <- function(m) {
  sample(0:k_star, size = m, prob = p_star, replace = TRUE) /
  (k_star + 1) +
    runif(m, 0, 1/(k_star + 1))
}

# MLE
mle <- function(x, k) {
  p <- rep(NA, k + 1)
  for (i in 1:(k + 1)) {
    p[i] <- max(sum(x < i / (k + 1) & x >= (i - 1) / (k + 1)) /
```

```
    length(x), 1e-10)
    # Ensuring that probabilities are never 0
  }
  return(p)
}

# The density f_k
f <- function(x, p, k) {
  z <- rep(0, length(x))
  for (j in 1:length(x)) {
    for (i in 1:(k + 1)) {
      if (x[j] < i / (k + 1) & x[j] >= (i - 1) / (k + 1)) {
        z[j] <- max(p[i] * (k + 1), 1e-10)
        #Ensuring that density is never 0
      }
    }
  }
  return(z)
}

# Log-likelihood computation
log_likelihood <- function(x, k) {
  sum(log(f(x, mle(x, k), k)))
}

# Cross-entropy
cross_entropy <- function(x, k) {
  f_true <- function(x) {
    for (i in 1:(k_star + 1)) {
```

```
      if (x >= (i-1)/(k_star + 1) & x < i/(k_star + 1)) {
        return(p_star[i] * (k_star + 1))
      }
    }
    return(0)
  }

  p_hat <- mle(x, k)
  f_est <- function(x) {
    for (i in 1:(k + 1)) {
      if (x >= (i-1)/(k + 1) & x < i/(k + 1)) {
        return(p_hat[i] * (k + 1))
      }
    }
    return(1e-10)
  }

  # Numerical integration
  n_points <- 1000
  x_vals <- seq(0, 1, length.out = n_points)
  dx <- 1/n_points
  integral <- sum(sapply(x_vals, function(x) {
    ft <- f_true(x)
    fe <- f_est(x)
    if (ft > 0 && fe > 0) ft * log(fe) * dx else 0
  }))

  -integral
}
```

```r
# Simulation parameters
n <- 50          # Sample size
m <- 50          # Maximum model complexity (k)
n_sim <- 100     # Number of simulations

# Initialize storage
loglik <- matrix(NA, nrow = n_sim, ncol = m)
aic <- matrix(NA, nrow = n_sim, ncol = m)
Hstar <- matrix(NA, nrow = n_sim, ncol = m)

# Run simulations
for (sim in 1:n_sim) {
  x <- simulate(n)
  for (k in 1:m) {

    loglik[sim, k] <- log_likelihood(x, k)
    aic[sim, k] <- 2*k - 2*log_likelihood(x, k)
    Hstar[sim, k] <- cross_entropy(x, k)
  }
}

# Maximum Log-Likelihood Plot
plot(colMeans(loglik),
     type = "b", pch = 19, col = "darkblue",
     ylim = range(loglik, na.rm = TRUE),
     xlab = "Model Complexity (k)", ylab = "Average Log-Likelihood")
abline(v = k_star, lty = 2, col = "red")
grid()
```

```
# AIC Plot
plot(colMeans(aic),
     type = "b", pch = 19, col = "darkgreen",
     ylim = range(aic, na.rm = TRUE),
     xlab = "Model Complexity (k)", ylab = "Average AIC")
abline(v = k_star, lty = 2, col = "red")
grid()

# Cross-Entropy Plot
plot(colMeans(Hstar),
     type = "b", pch = 19, col = "purple",
     ylim = range(Hstar, na.rm = TRUE),
     xlab = "Model Complexity (k)",
     ylab = expression("Average H"["k"]^"*"))
abline(v = k_star, lty = 2, col = "red")
grid()
```

Code 3: Cross Entropy and Bias Estimates

```
library(ggplot2)

# Simulation data generated
simulate_data <- function(k_star,p_star, n){
  sample(0:k_star, size = n, prob = p_star, replace = TRUE) /
  (k_star + 1) + runif(n, 0, 1 / (k_star + 1))
}

# MLE
```

```r
compute_mle <- function(x, k) {
  p_hat <- numeric(k + 1)
  for (i in 1:(k + 1)) {
    p_hat[i] <- sum(x >= (i-1)/(k+1) & x < i/(k+1))/length(x)
  }
  p_hat <- pmax(p_hat, 1e-10)
  p_hat/sum(p_hat)
}

# Log-likelihood
log_likelihood <- function(x, k) {
  p_hat <- compute_mle(x, k)
  sum(log((k + 1) * p_hat[findInterval(x, seq(0, 1, length.out=k+2))]))
}



#Naive Estimate
naive_estimate <- function(x, k) {
  -log_likelihood(x, k) / length(x)
}



# AIC Bias-Corrected Estimate
aic_estimate <- function(x, k) {
  (k/length(x))+naive_estimate(x,k)
}

# Naive MLE Estimate
mle_estimate <- function(x, k) {
```

```
  p_hat <- compute_mle(x, k)
  -sum(p_hat * log(p_hat)) - log(k + 1)
}


# Miller-Madow Bias-Corrected Estimate
millermadow_estimate <- function(x, k) {
  p_hat <- compute_mle(x, k)
  m <- sum(p_hat > 0)  # Number of non-zero bins
  entropy_mm <- (-sum(p_hat * log(p_hat))) - (m - 1)/(2*length(x))
  entropy_mm - log(k + 1)
}

# Jackknife Estimate
jackknife_estimate <- function(x, k) {
  n <- length(x)
  full_est <- naive_estimate(x, k)

  loo_ests <- sapply(1:n, function(i) {
    naive_estimate(x[-i], k)
  })

  n * full_est - (n - 1) * mean(loo_ests)
}

# Run simulations
run_simulation <- function(k_star, p_star, n_values = seq(10, 400,
by=10), nsim=100) {
  results <- data.frame()
```

```
for (n in n_values) {
  true_vals <- numeric(nsim)
  naive_vals <- numeric(nsim)
  aic_vals <- numeric(nsim)
  mle_vals <- numeric(nsim)
  mm_vals <- numeric(nsim)
  jack_vals <- numeric(nsim)

  for (i in 1:nsim) {
    x <- simulate_data(k_star,p_star, n)
    k <- k_star  # Estimating for true model complexity

    # True cross-entropy
    p_hat <- compute_mle(x, k)
    true_vals[i] <- -sum(p_star * log(p_hat)) - log(k + 1)

    aic_vals[i] <- aic_estimate(x, k)
    mle_vals[i] <- mle_estimate(x, k)
    mm_vals[i] <- millermadow_estimate(x, k)
    jack_vals[i] <- jackknife_estimate(x, k)
  }

  results <- rbind(results, data.frame(
    n = n,
    true = mean(true_vals),
    aic = mean(aic_vals),
    mle = mean(mle_vals),
    millermadow = mean(mm_vals),
```

```
      jackknife = mean(jack_vals)
    ))
  }
  return(results)
}
u <- c(10, 1, 1, 10)
k_star <- length(u) - 1
p_star <- u / sum(u)


results=run_simulation(k_star,p_star,nsim=100)

# Cross-Entropy Estimates Plot
ggplot(results, aes(x = n)) +
  geom_line(aes(y = true, color = "True"), linewidth = 1) +
  geom_line(aes(y = aic, color = "AIC")) +
  geom_line(aes(y = mle, color = "MLE")) +
  geom_line(aes(y = millermadow, color = "Miller-Madow")) +
  geom_line(aes(y = jackknife, color = "Jackknife")) +
  labs(x = "Sample Size", y = "Cross-Entropy",
       color = "Estimator") +
  theme_minimal()

# Bias visualization plot
results_bias <- data.frame(
  n = results$n,
  aic = results$n * (results$aic - results$true),
  mle = results$n * (results$mle - results$true),
  millermadow = results$n * (results$millermadow - results$true),
```

```
  jackknife = results$n * (results$jackknife - results$true)
)

ggplot(results_bias, aes(x = n)) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "gray")
+
  geom_line(aes(y = aic, color = "AIC")) +
  geom_line(aes(y = mle, color = "MLE")) +
  geom_line(aes(y = millermadow, color = "Miller-Madow")) +
  geom_line(aes(y = jackknife, color = "Jackknife")) +
  labs(x = "Sample Size (n)",
       y = "n    Bias",
       color = "Estimator") +
  theme_minimal()
```

Code 4: Asymptotic behavior: $P(A_n)$

```
library(nloptr)
library(ggplot2)
library(dplyr)

# Binary entropy function h
h <- function(x) {
  -x * log(x) - (1 - x) * log(1 - x)
}

# Inverse of h using numerical optimization
h_inv <- function(y) {
```

```
  obj_fn <- function(x) (h(x) - y)^2
  result <- nloptr(
    x0 = 0.5,
    eval_f = obj_fn,
    lb = 0,
    ub = 0.5,
    opts = list(algorithm = "NLOPT_LN_COBYLA", xtol_rel = 1e-8)
  )

  if (result$status < 0) return(NA)
  result$solution
}

# Function to calculate P(A_n)
P_A_n <- function(n) {
  if (n <= 0) return(NA)
  k <- n * h_inv(log(2) - 1/n)
  1 - 2 * pbinom(floor(k), size = n, prob = 0.5)
}

# Plot P(A_n)
n_values <- seq(10, 10000, by = 100)
P_values <- sapply(n_values, P_A_n)

df_P <- data.frame(n = n_values, P = P_values)

ggplot(df_P, aes(x = n, y = P)) +
  geom_line(color = "blue") +
  labs(x = "n", y = expression(P(A[n]))) +
```

```
  theme_minimal()
```

Code 5: Asymptotic behavior: $P(A_n|t)$

```
library(nloptr)
library(ggplot2)
library(dplyr)

# Function f(p|t)
f <- function(p, t) {
  if (p == 0) {
    term1 <- 0
  } else {
    term1 <- p * log(p / t)
  }

  if (p == 1) {
    term2 <- 0
  } else {
    term2 <- (1 - p) * log((1 - p) / (1 - t))
  }

  return(term1 + term2)
}

# f1 inverse (for p in (0,t])
f1_inverse <- function(y, t, lower = 1e-10, upper = t - 1e-10) {
  if (y <= 0) return(0)
```

```
  # Objective function
  obj <- function(p) {
    (f(p, t) - y)^2
  }

  # Use optimization to find the inverse
   result <- nloptr::bobyqa(x0 = t/2, fn = obj, lower = lower,
   upper = upper)

  return(result$par)
}

# f2 inverse (for p in (t,1))
f2_inverse <- function(y, t, lower = t + 1e-10, upper = 1 - 1e-10) {
  if (y <= 0) return(1)

  obj <- function(p) {
    (f(p, t) - y)^2
  }

  result <- nloptr::bobyqa(x0 = (t + 1)/2, fn = obj, lower = lower,
  upper = upper)

  return(result$par)
}

# Calculation of P(A_n|t)
P_A_given_t <- function(n, t) {
  y <- 1/n
```

```
  # Handle edge cases
  if (y >= f(0, t) && y >= f(1, t)) {
    return(1)  # All p values satisfy f(p|t) < y
  }

  # Find the lower and upper bounds
  p_lower <- tryCatch(f1_inverse(y, t), error = function(e) 0)
  p_upper <- tryCatch(f2_inverse(y, t), error = function(e) 1)

  # Calculate the binomial probabilities
  k_lower <- ceiling(n * p_lower)
  k_upper <- floor(n * p_upper)

  # Ensure we don't go outside the valid range
  k_lower <- max(0, k_lower)
  k_upper <- min(n, k_upper)

  if (k_lower > k_upper) {
    return(0)
  }

  # Calculate the probability using binomial CDF
  prob <- pbinom(k_upper, size = n, prob = t) - pbinom(k_lower - 1,
  size = n, prob = t)

  return(prob)
}
```

```r
# Simulation function to study P(A_n|t) for different n and t
simulate_P_A <- function(n_values = seq(10, 10000, by = 10),
                         t_values = seq(0.1, 0.9, by = 0.1)) {
  results <- expand.grid(n = n_values, t = t_values)

  # Calculate P(A_n|t) for each combination
  results$P <- mapply(P_A_given_t, results$n, results$t)

  return(results)
}

# Run simulations
set.seed(123)  # For reproducibility
sim_results <- simulate_P_A()

# Plot P(A_n|t) vs n for different t values
ggplot(sim_results, aes(x = n, y = P, color = as.factor(t))) +
  geom_line() +
  labs(x = "Sample size (n)",
       y = "P(A_n|t)",
       color = "t value") +
  theme_minimal()
```

Non-exclusive licence to reproduce thesis and make thesis public

I, Farhad Guliyev,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright, Model Selection and AIC, supervised by Joonas Sova.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Farhad Guliyev
21/05/2025