University of Minnesota Duluth
Swenson College of Science and Engineering
Mathematics & Statistics Department

# Variations of the EM Algorithm in Finite Mixture Models

Final Project

Duluth, 2021

Jan Matas

# Abstract

Finite mixture models provide a convenient framework for model-based clustering. Traditionally, the model parameters are estimated by maximum likelihood estimation, fulfilled by the expectation-maximization (EM) algorithm. Such approach to clustering has many advantages but also several pitfalls. Some of those issues can be overcome by varying the EM algorithm. We describe two variants of the EM algorithm, namely the Classification EM (CEM) and the Stochastic EM (SEM).

We study the performance of the standard EM, CEM, and SEM measured by the Adjusted Rand index in simulation studies for two different mixtures. First, we examine a finite Gaussian mixture model which is, by far, the most popular and widely studied mixture model. Then, we present our results for a finite mixture of Markov chains. We conducted a simulation study similar to the one with Gaussian mixture but additionally, we studied how frequently the three procedures identify the correct number of components $K$.

Lastly, we study the historical outcomes of the U.S. Senate elections on the state level using a mixture of Markov models. By fitting a mixture of Markov models to the data we collected, we first cluster the U.S. states. The resulting clusters resemble the well-known red and blue states partitioning but also other clusters were identified with the increasing number of the mixture components. We also show how to use the fitted model to perform a forecast for future elections. Specifically, we provide a forecast for 2022 Senate elections.

**Keywords**: Finite mixture models, clustering, EM algorithm, classification EM, stochastic EM, adjusted Rand Index.

# Acknowledgments

I would like to express my deep gratitude to Dr. Igor Melnykov who supervised this final project. I appreciate his scientific guidance, valuable comments, and his patience and support during my work on research. It was truly a pleasure to study the cutting-edge topic of model-based clustering under him. Also, his insight into the United States' politics contributed greatly to the interpretation of the results.

I would also like to thank Dr. Dalibor Fronček who inspired me to go abroad and study in the United States, to Dr. Xuan Li for serving on my defense committee, and to the whole Department of Mathematics and Statistics at the University of Minnesota Duluth. After the warm reception at the very beginning, I met so many wonderful professors and friends there. I am thankful for the opportunity to be immersed in a new culture for the duration of my degree. This has allowed me to grow as a person, and the UMD and Duluth will continue to hold a special place in my heart.

Next, my sincere thanks go to Dr. Petr Stehlík from the University of West Bohemia who filled me with enthusiasm for mathematics at the very beginning of my academic journey and was always incredibly supportive. I am also deeply thankful to Dr. Jan Pospíšil, also from the University of West Bohemia, who introduced me to the exciting world of mathematical modeling in finance.

Lastly, I want to thank to my family whose value to me only grows with age and to my future wife, Dominika, for her continuous loving support.

# Contents

# Chapter 1

# Introduction to Finite Mixture Models

Let $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$ be independent, identically distributed $p$-dimensional observations from a distribution with probability density function

$$f(\boldsymbol{x}; \boldsymbol{\pi}) = \sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x}), \tag{1.1}$$

where $K$ is the number of mixture components and $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)^T$ represents the vector of mixing proportions. Then, $\pi_k$ can be interpreted as the probability that the observation $\boldsymbol{X}_i$ belongs to the $k$th subpopulation with corresponding density $f_k(\boldsymbol{x})$ called the $k$th component density. That means that $\boldsymbol{\pi}$ lies in a $(K-1)$-dimensional simplex with $0 \leq \pi_k \leq 1$ for every $k = 1, \ldots, K$ and $\sum_{k=1}^{K} \pi_k = 1$.

Next, assume that the functional form of each $f_k$ is completely known, i.e. $f_k(\boldsymbol{x}) = f_k(\boldsymbol{x}; \boldsymbol{\vartheta}_k)$, where $\boldsymbol{\vartheta}_k$ is a vector of parameters. Then, we can rewrite (1.1) as

$$f(\boldsymbol{x}; \boldsymbol{\vartheta}) = \sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x}; \boldsymbol{\vartheta}_k), \tag{1.2}$$

where $\boldsymbol{\vartheta} = (\boldsymbol{\pi}^T, \boldsymbol{\vartheta}_1^T, \ldots, \boldsymbol{\vartheta}_K^T)^T$. We refer to (1.2) as a finite mixture model density with $K$ components and parameter vector $\boldsymbol{\vartheta}$. Fitting such a model to data means that we need to estimate all the parameters in $\boldsymbol{\vartheta}$ when $K$ is known. If $K$ is not provided, we need to estimate $K$ as well.

Finite mixture models evolved and gained significant popularity with rising computational power of computers and nowadays, techniques involving such models are used for modeling non-homogeneous populations as well as for model-based clustering, which has significant applications in machine learning. Nevertheless, mixture distributions are not a brand new idea.

According to Melnykov and Maitra [18], the first historical record of finite mixture models can be found in an article from 1886 by Newcomb [20], who used them in the context of modeling outliers, although common reference is made to Pearson [21] who was the first author who explicitly addressed the decomposition problem in characterizing sub-populations of a general population. He used a mixture of two normal distributions to model ratios of forehead lengths to body lengths in a shore crab population. The motivation for this work was that asymmetry in the histogram of these ratios could signal evolutionary divergence. Although Pearson was successful in identifying two potentially distinct sub-populations, the estimation of the model parameters using the method of moments proved to be complicated and, in fact, infeasible for mixtures of more than two normal distributions. This is the reason why mixture models gained popularity relatively recently with refinement of the EM algorithm which was described as a general approach for problems with missing data in 1977 by Dempster et al. [7]. We will introduce the EM algorithm later, in Section 1.2.

More recently, mixtures of Poisson distributions have been used in positron emission tomography [25] or document classification in the context of information retrieval [13]. Nevertheless, the most popular and widely used mixture models are those with Gaussian components [15, 26, 3]. For a comprehensive survey on the history and applications of finite mixture models, we refer the reader to [16].

## 1.1 Model-based Clustering

Finite mixture models provide a convenient framework for model-based clustering. Clustering or cluster analysis is the task of grouping a set of objects in such a way that objects in the same group, called a cluster, are more similar to each other than to those in other groups. There are many frameworks and heuristics for clustering based on distance among observations, calculating centroids, etc. The approach when we assume that each group has its own distribution and a corresponding probability representation is called model-based or distribution-based clustering.

If the $k$th group is represented by $f_k(\boldsymbol{x}; \boldsymbol{\vartheta}_k)$ and $\pi_k$ is the inclusion probability, i.e., the probability that a random observation comes from the $k$th group, we can assume that $\boldsymbol{X}_1, \boldsymbol{X}_2, \dots, \boldsymbol{X}_n$ are a sample from (1.2). The most common choice of $f_k$'s is by far multivariate normal densities but in this work, we also consider probability mass functions representing categorical sequences modeled by Markov chains. Nevertheless, regardless of the specific choices of $f_k$'s, we can fit a mixture model to a data set and assign each observation to a predefined number of $K$ groups. Since we can think of the inclusion probabilities (or mixing proportions) $\boldsymbol{\pi}$, as prior probabilities that an observation comes from a specific mixing distribution, the Bayes rule provides an elegant way of assigning the group membership. Thus, every observation is assigned to the group having the highest posterior probability that an observation originated from this group. Denoting the possible groups by $C_1, \dots, C_k$, we are interested in

$$\pi_{ik} = \mathbb{P}[\boldsymbol{x}_i \in C_k | \boldsymbol{X}_i = \boldsymbol{x}_i] = \frac{\mathbb{P}[\boldsymbol{X}_i = \boldsymbol{x}_i | \boldsymbol{x}_i \in C_k]\mathbb{P}[\boldsymbol{x}_i \in C_k]}{\mathbb{P}[\boldsymbol{X}_i = \boldsymbol{x}_i]},$$

which can be rewritten as

$$\pi_{ik} = \frac{\pi_k f_k(\boldsymbol{x}_i; \boldsymbol{\vartheta}_k)}{\sum_{k'=1}^{K} \pi_{k'} f_{k'}(\boldsymbol{x}_i; \boldsymbol{\vartheta}_{k'})}. \tag{1.3}$$

Since the denominator is always the same for a given observation, choosing the highest posterior probability is equivalent to finding the group index corresponding to the biggest value $\pi_k f_k(\boldsymbol{x}_i; \boldsymbol{\vartheta}_k), k = 1, 2, \dots, K$.

## 1.2 Parameter Estimation Using the EM Algorithm

The traditional method of estimating the parameters $\boldsymbol{\vartheta}$ of model (1.2) is maximum likelihood estimation fulfilled by the expectation-maximization (EM) algorithm [7]. The EM algorithm is an iterative numerical procedure that consists of two steps. The first step is called E-step (expectation) and the second step is called M-step (maximization).
Using mathematical notation, we would like to maximize the likelihood function $L(\boldsymbol{\vartheta}) = L(\boldsymbol{\vartheta}; \boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$ with respect to the vector of parameters $\boldsymbol{\vartheta}$. Then, we call

$$\widehat{\boldsymbol{\vartheta}} = \arg\max_{\boldsymbol{\vartheta}} L(\boldsymbol{\vartheta})$$

the maximum likelihood estimate (MLE) of $\boldsymbol{\vartheta}$. Following the definition of a likelihood function as the product of a model density function evaluated at given observations, we get

$$L(\boldsymbol{\vartheta}) = \prod_{i=1}^{n} f(\boldsymbol{x}_i; \boldsymbol{\vartheta}),$$

where in general, $f$ is a probability density function. In our case $f$ is of form (1.2), hence we can further rewrite the likelihood function as

$$L(\boldsymbol{\vartheta}) = \prod_{i=1}^{n} \sum_{k=1}^{k} \pi_k f_k(\boldsymbol{x}_i; \boldsymbol{\vartheta}_k). \tag{1.4}$$

Unfortunately, the likelihood function $L(\boldsymbol{\vartheta})$ is usually very complicated and often infeasible to optimize directly for it is generally not convex. The basic principle of the EM algorithm to deal with the complexity of optimization of likelihood functions of models for incomplete data is to replace the maximization of $L(\boldsymbol{\vartheta})$ with maximization of the $Q$-function which is the conditional expectation of $L(\boldsymbol{\vartheta})$ given that we assume a complete data structure.

Let $\mathcal{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n\}$ be the observed data. Recall that the parameter $\pi_k$ in (1.2) can be interpreted as the probability that a random observation comes from the $k$th group. If we knew the actual partitioning of the observations into $K$ groups, we could simply compute $\pi_k$ as the frequency of observations belonging to the $k$th group and fit an individual model represented by a probability density function $f_k$ to the observations in the $k$th group for every $k = 1, \dots, K$. Although we do not know the actual partitioning, it still makes sense to assume that there is some and that we can, in some way, assign a given observation into one of the $K$ groups. We can think of the group membership of the $i$th observation as a random variable $Z_i$. For example, $Z_i = 2$ means that the $i$th observation belongs to group 2. Knowing the group label $Z_i$ for every observation $\boldsymbol{x}_i, i = 1, \dots, n$, transforms our incomplete data $\mathcal{X}$ into complete data $\mathcal{X}_C = \{(\boldsymbol{x}_1, z_1), (\boldsymbol{x}_2, z_2), \dots, (\boldsymbol{x}_n, z_n)\}$.
The $Q$-function is constructed by assuming that the labels $z_1, \dots, z_n$ are known. In that case, we can rewrite the likelihood function (1.4) in terms of complete data $\mathcal{X}_C$ as

$$L_C(\boldsymbol{\vartheta}) = \prod_{i=1}^{n} \prod_{k=1}^{K} \pi_k f_k(\boldsymbol{x}_i; \boldsymbol{\vartheta}_k)^{I[Z_i = k]},$$

where $I[\cdot]$ represents the indicator function. We refer to $L_C$ as the complete-data likelihood function. Furthermore, the logarithm of likelihood function is often maximized for its more convenient summation form rather than the original one involving products. Since logarithmic transformation is injective, the values maximizing the log-likelihood function are the same as in case of the original likelihood function. Then, the complete-data log-likelihood function can be written as

$$\ell_C(\boldsymbol{\vartheta}) = \sum_{i=1}^{n} \sum_{k=1}^{K} I[Z_i = k](\log \pi_k + \log f_k(\boldsymbol{x}_i; \boldsymbol{\vartheta}_k)). \tag{1.5}$$

Finally, the $Q$-function is defined as the expectation of the complete log-likelihood function:

$$Q(\boldsymbol{\vartheta}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \pi_{ik}(\log \pi_k + \log f_k(\boldsymbol{x}_i; \boldsymbol{\vartheta}_k)), \tag{1.6}$$

where $\pi_{ik}$ represents the posterior probability that $\boldsymbol{x}_i$ came from the $k$th group which arose from $\mathbb{E}[I[Z_i = k]] = \mathbb{P}[Z_i = k]$. We will often denote the $Q$-function by $Q(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}_b)$, emphasizing the fact that to compute the posterior probabilities by formula (1.3), we need to have some values of the model parameters. This brings us to the core principle of the iterative EM algorithm.
Given a current parameter estimate $\boldsymbol{\vartheta}_b$:

- Expectation step: find the conditional expectation of the complete-data log-likelihood function $Q(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}_b)$ which includes computing the posterior probabilities $\pi_{ik}$ based on the given parameters $\boldsymbol{\vartheta}_b$.

- Maximization step: update the parameter estimate by maximizing $Q(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}_b)$, i.e.,

$$\boldsymbol{\vartheta}_{b+1} = \arg\max_{\boldsymbol{\vartheta}} Q(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}_b).$$

By the nature of the E-step, it is guaranteed that the sequence of parameter estimates resulting from the EM algorithm will lead to a non-decreasing sequence of log-likelihood values. However, there are several practical issues that need to be accounted for. The main cause of the issues with the EM algorithm is the fact that the likelihood function of a mixed model is typically a very complicated, non-convex, multi-modal function, hence an always-climbing iterative algorithm may not be able to find the global maximum. In fact, different initial values of the parameter estimates $\boldsymbol{\vartheta}_0$ may result in different outcomes of the algorithm. In some particular cases, the outcome of the EM algorithm may be very sensitive to the choice of $\boldsymbol{\vartheta}_0$. Another issue is that the numerical procedure can be time-consuming, especially when the number of components in the mixture is large and we have a large data set. In such cases, we need to consider the trade-off between accuracy and speed.

## 1.3 Variations of the EM Algorithm

There are many modifications of the EM algorithm that aim to account for the aforementioned issues. In this work, we consider the Classification EM (CEM) and Stochastic EM (SEM) variants of the algorithm which we compare in simulation studies together with the standard EM based on the quality of fitting mixtures of Gaussian densities in Section 2.2 and mixtures of Markov chains in Section 3.2.

### 1.3.1 Classification EM

In the Classification EM [6], we directly estimate the unknown group memberships using the posterior probabilities, i.e., the estimate of the group membership of the $i$th observation is the number of the group having the highest posterior probability that the observation originated from this group. Thus, the expectation step is replaced by estimation of the group memberships, which we can call the Classification step. Denoting $\mathbf{Z} = (Z_1, \ldots, Z_n)^T$ the vector of unknown group memberships and given a current parameter estimate $\boldsymbol{\vartheta}_b$, CEM consists of three steps:

- Expectation step: compute the posterior probability $\pi_{ik}$ for every $i = 1, \ldots, n$, and $k = 1, \ldots, K$.
- Classification step: find the estimate of $Z_i$ by $\widehat{z}_i = \arg\max_k \pi_{ik}$ for every $i = 1, \ldots, n$.
- Maximization step: find $\boldsymbol{\vartheta}_{b+1} = \arg\max_{\boldsymbol{\vartheta}} Q(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}_b)$.

The advantage of CEM is its faster and easier implementation because we do not have to derive the Q-function by conditioning the log-likelihood function. Instead, the M-step consists of finding the MLE of the parameters of the $k$th mixing model from the observations assigned to the $k$th group based on the estimated group memberships $\mathbf{Z}$. That means that there is still some function $Q(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}_b)$ that is being maximized instead of the actual log-likelihood function but we do not have the exact formula to represent it. Also, since that Q-function is not the expectation of the complete-data log-likelihood as in the case of the standard EM, the monotonicity property of the EM algorithm is lost which may, in fact, cause troubles stopping the algorithm using solely the distance between two consecutive estimates as the stopping criterion. Another advantage of CEM is its fast pace of convergence, especially in the very first couple of steps. This property of CEM comes forward to be exploited by running CEM in the initial stage of fitting a mixture model and after that, running the standard EM.

### 1.3.2 Stochastic EM

The stochastic EM [5] shares the idea of directly estimating the group memberships with CEM, but in addition, it introduces randomness by not taking directly the memberships with the highest posterior probabilities but instead, it randomly samples the group index from the distribution represented by the posterior probabilities, i.e., given a current parameter estimate $\boldsymbol{\vartheta}_b$:

- Expectation step: compute the posterior probability $\pi_{ik}$ for every $i = 1, \dots, n$, and $k = 1, \dots, K$.
- Simulation step: estimate $Z_i$ by drawing a sample from the discrete distribution characterized by $\pi_{ik}$ for every $i = 1, \dots, n$.
- Maximization step: find $\boldsymbol{\vartheta}_{b+1} = \arg\max_{\boldsymbol{\vartheta}} Q(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}_b)$.

Similarly to CEM, SEM is easier to implement and should offer faster convergence but the monotonicity is not retained. However, the main motivation behind introducing randomness to the optimization procedure is to avoid getting stuck in a local maximum, thus improving the global convergence properties of the algorithm.

## 1.4 Model Selection and Clustering Comparison

In practical applications, we often face a problem of comparing different models for a given data set. When we are interested in comparing the explanatory ability of several models, one approach is to compare their likelihoods for given data. Then, we would choose the model with maximum likelihood since it fits the data best. However, such comparison does not account for the number of parameters of a model. In fact, we usually prefer models with as few parameters as possible. If two models provide a comparable fit for a given data set, we would prefer the one with fewer parameters because it is simply less complicated and models with large numbers of parameters tend to overfit data which is not desirable. An overfitted model describes a particular data set in too much detail and as a result, the model may not be valid anymore after adding new data or it can perform very poorly in predicting future observations. In fact, the essence of overfitting is including some of the residual variation in the model in hand as if that variation represented the underlying model structure. Hence, also the explainability of overfitted models usually suffers greatly.

Considering the importance of the number of parameters of a model together with its fit for given data, another approach to compare different models is based on information criteria. Based on the discussion in [17], we will use the Bayes Information Criterion (BIC) which is the most popular choice in the mixture modeling framework. The BIC [23] is formally defined as

$$\text{BIC} = M \ln(N) - 2\ln(\widehat{L}),$$

where $M$ is the number of parameters estimated in the model, $N$ is the sample size, and $\widehat{L}$ is the maximized value of the likelihood function of the model for given data.

In case we know the true memberships of observations, for example when we are conducting a simulation study, we can use other metrics to evaluate the performance of a model. In particular, we can utilize the known memberships to measure how accurately the model memberships match the true memberships. Nevertheless, computing simply the ratio of correctly classified observations to the total number of observations is not sufficient because such a measure is not label invariant. For example, consider a data set where the observed data fall into exactly two apparently distinct clusters. If we choose labels 1, 2 to denote the membership of each observation, there are still two ways to assign the labels depending on for which cluster we decide to use label 1 and for which label 2. To overcome this issue, we need to use a label invariant measure. A popular choice in the model-based clustering framework is the Adjusted Rand Index introduced by Hubert and Arabie [9] which is an extension of the Rand Index named after Rand [22].

Adjusted Rand Index ($\mathcal{AR}$) measures agreement between two data partitions representing distinct groups of observations. We do not have to know the actual class memberships to compute $\mathcal{AR}$ but if the memberships are available, $\mathcal{AR}$ can be used as an invariant measure of agreement of the model clusters with the actual groups. In that case, the zero value of $\mathcal{AR}$ represents the level of agreement with a completely random allocation of points into groups. The upper bound of $\mathcal{AR}$ is one which is achieved when two partitionings match entirely. Values smaller than one indicate lower agreement.

# Chapter 2

# Gaussian Finite Mixture Model

In this chapter, we introduce the Gaussian (normal) mixture which is by far the most popular mixture model. Then, we present the results of the simulation study that we conducted to empirically compare the standard EM, CEM, and SEM introduced in Section 1.2.

## 2.1 Mixture of Normal Distributions and Parameter Estimation

Recall the general mixed model introduced at the beginning of Chapter 1 and assume that the $k$th subpopulation follows a $p$-variate normal distribution with mean vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$, a distribution which we denote $N_p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. Then, the $k$th subpopulation density is given by

$$\varphi(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = (2\pi)^{-\frac{p}{2}} |\boldsymbol{\Sigma}_k|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_k)\right\}$$

and the general model (1.2) can be rewritten as

$$f(\boldsymbol{x}; \boldsymbol{\vartheta}) = \sum_{k=1}^{K} \pi_k \varphi(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k).$$

Next, the likelihood function for a sample of observations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ follows the form

$$L(\boldsymbol{\vartheta} | \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \varphi(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k).$$

However, such a function is often very difficult to maximize. The standard way to handle this is to use the EM algorithm introduced in Section 1.2. We assume that the unavailable group memberships $z_1, \dots, z_n$ are known. Then we obtain the complete-data likelihood function in the form

$$L_C(\boldsymbol{\vartheta}|\boldsymbol{x}_1, \dots, \boldsymbol{x}_n) = \prod_{i=1}^{n} \prod_{k=1}^{K} (\pi_k \varphi(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k))^{I[Z_i=k]}, \tag{2.1}$$

where $I[\cdot]$ represents the indicator function. By taking expectation with respect to the group memberships, the corresponding Q-function can be derived:

$$\begin{aligned} Q(\boldsymbol{\vartheta}|\boldsymbol{x}_1, \dots, \boldsymbol{x}_n) = &-\frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{K} \pi_{ik} \{\log |\boldsymbol{\Sigma}_k| + (\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k)\} \\ &+ \sum_{i=1}^{n} \sum_{k=1}^{K} \pi_{ik} \log \pi_k - \frac{pn}{2} \log 2\pi, \end{aligned}$$

where $\pi_{ik}$ is the posterior probability that the $i$th observation belongs to the $k$th subpopulation. Then, the E-step of the EM algorithm consists of updating the posterior probabilities given the current parameter estimates $\boldsymbol{\vartheta}^{(b-1)}$ by

$$\pi_{ik} = \frac{\pi_k^{(b-1)} \varphi(\boldsymbol{x}_i; \boldsymbol{\mu}_k^{(b-1)}, \boldsymbol{\Sigma}_k^{(b-1)})}{\sum_{k'=1}^{K} \pi_{k'}^{(b-1)} \varphi(\boldsymbol{x}_i; \boldsymbol{\mu}_{k'}^{(b-1)}, \boldsymbol{\Sigma}_{k'}^{(b-1)})}.$$

Maximizing the Q-function, we get the formulas for updating the parameter estimates in the M-step:

$$\pi_k^{(b)} = \frac{1}{n} \sum_{i=1}^{n} \pi_{ik}^{(b)}, \quad \boldsymbol{\mu}_k^{(b)} = \frac{\sum_{i=1}^{n} \pi_{ik}^{(b)} \boldsymbol{x}_i}{\sum_{i=1}^{n} \pi_{ik}^{(b)}}, \quad \boldsymbol{\Sigma}_k^{(b)} = \frac{\sum_{i=1}^{n} \pi_{ik}^{(b)} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(b)})(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(b)})^T}{\sum_{i=1}^{n} \pi_{ik}^{(b)}}.$$

## 2.2 Simulation Study

The goal of the simulation study we conducted was to examine and compare the performance of the EM algorithm and its variants detecting clusters in the data via fitting a finite mixture with Gaussian components. In particular, we studied the quality of the fits and corresponding clusterings achieved by the standard EM, CEM, and SEM measured by the adjusted Rand index ($\mathcal{AR}$).

We arranged a family of specific scenarios in order to examine the performance of the three procedures. We assumed that there are $K$ mixture components following $p$-dimensional normal distributions from which a total of $n$ observations are drawn such that the average pairwise overlap of the groups equals a pre-defined level $\Omega$. To be more specific, pairwise overlap is defined as a sum of two misclassification probabilities. It measures the degree of interaction between components and can be readily employed to control the clustering complexity of datasets simulated from mixtures, as explained in the description of R package `MixSim` [19] which we used for generating data.

It can be anticipated that the introduction of a large number of mixing components will lead to harder scenarios because, for example, overlaps of more than two groups may arise. Also, a small sample size $n$ makes a scenario harder because the parameter estimates may not be accurate enough. On the contrary, larger $n$ should lead to more accurate estimates (this is guaranteed for the standard EM since in this case, the estimators are MLE hence consistent) which results in a more accurate partitioning (we can measure accuracy because we know the true class memberships of observations in synthetic data sets). Nevertheless, we focused mainly on examining the behavior of the three procedures when the number of dimensions $p$ and the average pairwise overlap $\Omega$ are varied.

It is known that model-based clustering techniques do not perform particularly well in high-dimensional spaces. This is mainly due to overparametrization. Assuming $K$ Gaussian components in a $p$-dimensional space, the total number of parameters to estimate is equal to $(K-1)+Kp+Kp(p-1)/2$, where $(K-1), Kp$, and $Kp(p-1)/2$ are respectively the numbers of free parameters for mixing proportions, the means, and the covariance matrices. Because the number of free parameters increases quadratically with respect to $p$, a large sample size is required for higher values of $p$. There are many approaches to deal with the problem of high-dimensional data (also known as the *curse of dimensionality*), such as dimension reduction, subspace clustering and other methods, as suggested, for example, by Bouveyron and Brunet-Saumard [4].

We studied the effect of the number of dimensions on the performance of the standard EM, CEM, and SEM. To design gradually challenging scenarios, we chose $p = 2, 5, 10$, and $\Omega = 0.01, 0.05, 0.10$ but because the simulations were time-demanding we decided to explore scenarios only for fixed $K = 10$ and evenly distributed mixing proportions $\pi_k = 0.1$. To ensure that there are enough points to estimate the model parameters of each component, we set $n = 5000$ (then approximately 500 points are drawn from each component). Figure 2.1 shows an example of such data sets for the three different values of $\Omega$.

For each $(p, \Omega)$-combination, we generated 50 data sets, each of size $n = 5000$, and ran the standard EM, CEM, and SEM. In all cases, we assumed that $K$ is known to be 10 and we used the relative distance between two subsequent values of the likelihood function as the stopping criterion. Generally, even when the average pairwise overlap $\Omega$ is small, outcomes of the EM procedure depend heavily on the initial parameter estimates, particularly the mean estimates. Inconvenient initialization may result in some clusters not being recognized at the expense of overfitting other clusters. This issue is depicted in Figure 2.2. To account for that, we implemented random initialization of means such that $K$ points are drawn from the data set and they are used as the initial mean estimates. For each data set, we initialized 50 means using this method. The codes implementing the simulation study in R are available on Github [14].

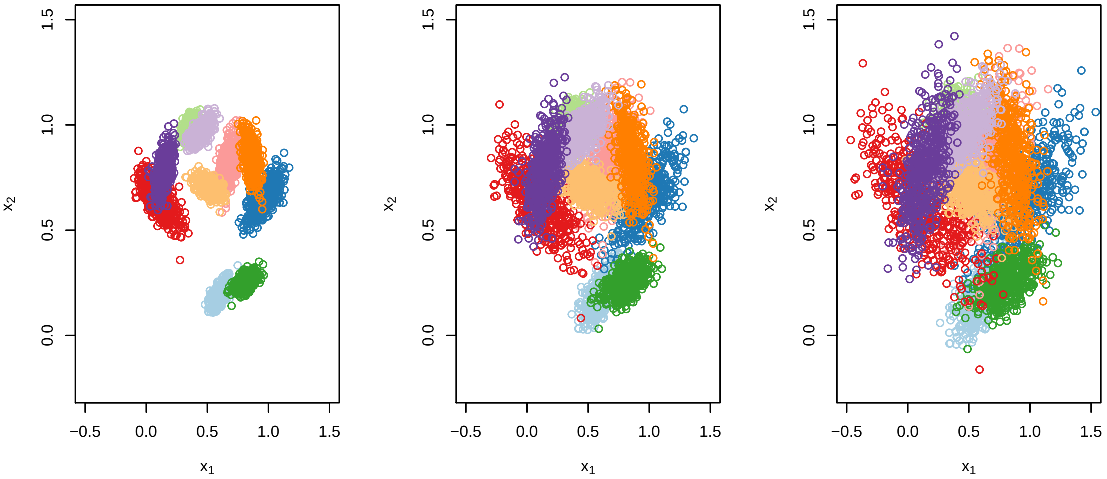


Figure 2.1: Comparison of data sets generated from a mixture of Gaussian components for different values of the average pairwise overlap $\Omega$. From left to right, the values of $\Omega$ are $0.01, 0.05$, and $0.10$.

The results of the simulation study are summarized in Table 2.1. Looking at the table, we can observe that the median $\mathcal{AR}$ decreases significantly with increasing $\Omega$ while increasing $p$ does not have such a significant effect on $\mathcal{AR}$. Comparing the standard EM, CEM, and SEM, the standard EM seems to be consistently the most accurate of the three while CEM seem to be the least accurate since it was outperformed by SEM in all scenarios. Also, it appears that CEM suffers more from increasing overlap $\Omega$ compared to the other methods. When clusters heavily overlap each other, there are a lot of points that are difficult to assign to groups. While the standard EM continues improving the fit with every iteration, CEM is generally faster but may fall into a loop when it encounters problematic points. Thus CEM progresses much faster at the beginning when the outlines of clusters are formed but may be difficult to terminate later by using the relative distance of two subsequent log-likelihoods as a stopping criterion. It will simply continue jumping back and forth for the points that are difficult to cluster which causes that the stopping criterion is never small enough to stop the procedure. On the contrary, SEM seems to be more stable when $p$ is increasing, similarly to the standard EM, while being still faster than the standard EM. Based on this observation, it would be advisable to start the clustering procedure with CEM or SEM for the first couple of steps when they converge fast and finish the process with EM which guarantees improving outcomes with every iteration and leads to more accurate solutions.

It is also worth to mention that the lowest levels of the median $\mathcal{AR}$ for the scenarios with

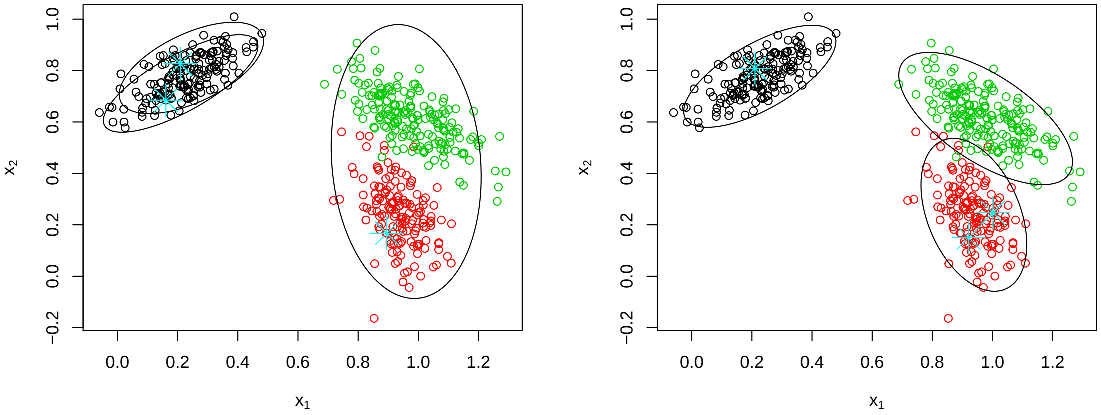


Figure 2.2: Outcomes of the EM procedure of fitting a Gaussian mixture are sensitive to the choice of initial mean estimates. The two figures illustrate the outcomes of fitting a mixture of three Gaussian components in a two-dimensional space. The initial mean estimates are depicted as stars and the ellipses represent the pdfs of the Gaussian mixing components. The figure on the left illustrates a case when an inconvenient initialization of the means resulted in a fitted model that does not recognize the clusters correctly. The figure on the right shows a mean initialization that resulted in proper recognition of all the three clusters.

| | | EM | | CEM | | SEM | |
|---|---|---|---|---|---|---|---|
| $p$ | $\Omega$ | med | IQR | med | IQR | med | IQR |
| 2 | 0.01 | 0.8399 | 0.0946 | 0.8581 | 0.0577 | 0.8665 | 0.0951 |
| 2 | 0.05 | 0.6513 | 0.0391 | 0.6110 | 0.0541 | 0.6479 | 0.0270 |
| 2 | 0.1 | 0.4706 | 0.0422 | 0.4042 | 0.0568 | 0.4656 | 0.0371 |
| 5 | 0.01 | 0.9092 | 0.0103 | 0.8894 | 0.0814 | 0.9075 | 0.0095 |
| 5 | 0.05 | 0.6654 | 0.0188 | 0.5768 | 0.0579 | 0.6628 | 0.0241 |
| 5 | 0.1 | 0.4744 | 0.0319 | 0.3361 | 0.0535 | 0.4239 | 0.0484 |
| 10 | 0.01 | 0.9110 | 0.0104 | 0.9045 | 0.0762 | 0.9092 | 0.0114 |
| 10 | 0.05 | 0.6694 | 0.0173 | 0.5564 | 0.0607 | 0.6582 | 0.0199 |
| 10 | 0.1 | 0.4710 | 0.0238 | 0.2473 | 0.0506 | 0.4199 | 0.0422 |

Table 2.1: Results of the simulation study comparing the quality of clustering achieved by the standard EM, CEM, and SEM in scenarios characterized by the number of dimensions $p$ and the average pairwise overlap between components $\Omega$. The number of components and the sample size was assumed to be fixed at 10 and 5000 respectively. For each $(p, \Omega)$-combination, 50 data sets where generated. The quality of the clustering achieved by the three procedures is measured by the median Adjusted Rand Index ($\mathcal{AR}$) and the variability of the outcomes among the 50 data sets by the interquartile range (IQR).

$\Omega = 0.1$ are around 0.47 for the standard EM which may seem low but it is important to remember that $\mathcal{AR}$ of value one is achieved when the model clusters agree completely with the true clusters and $\mathcal{AR}$ of value zero represents the agreement between a completely random assignment of points to groups and the true clusters. Considering the fact that scenarios with high average pairwise overlap can include overlaps of multiple clusters, as we can see in Figure 2.1, the accuracy of the clustering can be considered successful.

# Chapter 3

# Mixture of Markov Models

In this chapter, we are concerned with modeling of categorical sequences that originate from discrete probability distributions. First, we introduce Markov chains and their higher-order generalizations and their mixtures. Then, we present the results of a simulation study that compares the standard EM, CEM and SEM introduced in Section 1.2. In the last part, we fit a mixture of Markov chains to a U.S. Senate election data set and we use the fitted model to cluster the U.S. states and to forecast the results of 2022 senate elections.

## 3.1 Markov Chains

Let $\{X_n, n \in \mathbb{N}\}$ be a stochastic process that takes on values from its state space $S = \{s_1, \ldots, s_J\}$. We suppose that whenever the process is in state $s_i$, there is a fixed probability $P_{ij}$ that it will next be in state $s_j$, i.e., we suppose that

$$\begin{aligned} P_{ij} = \mathbb{P}[X_{n+1} = s_j | X_n = s_i, X_{n-1} = s_{i_{n-1}}, \ldots, X_1 = s_{i_1}, X_0 = s_{i_0}] = \\ = \mathbb{P}[X_{n+1} = s_j | X_n = s_i], \end{aligned} \tag{3.1}$$

for all states $s_{i_0}, s_{i_1}, \ldots, s_{i_{n-1}}, s_i, s_j$, and for every $n \geq 0$. Such a stochastic process is known as a Markov chain.

The probabilities $P_{ij}$ are called transition probabilities and it is convenient to organize them in a so called transition matrix $P = (P_{ij})_{i,j=1}^{J}$. Then, if $\alpha_0$ is a vector of probabilities representing the initial distribution of the states, the formula $\alpha_n = P^n \alpha_0$ gives us the probability distribution of states after $n$ transitions. For a comprehensive work on Markov chains, we refer to Çinlar [10].

In fact, the model described above is a first order Markov chain. It is worth mentioning that there are higher order Markov models that differ in how many terms the model "looks into the past". More formally, a $p$th order Markov model has the property

$$\mathbb{P}[X_n = x_n | x_{n-1}, x_{n-2}, \ldots, x_0] = \mathbb{P}[X_n = x_n | x_{n-1}, x_{n-2}, \ldots, x_{n-p}],$$

where $x_{n-1}, x_{i-2}, \ldots, x_0$ is the sequence of values realized by the stochastic process before $x_n$ is reached. In some cases, higher order Markov chains may serve better for describing the dynamic properties of highly autocorrelated categorical time series but with increasing the order, the model becomes more and more complex. It is well known that a higher order Markov model can be represented in terms of a first-order model with a higher number of states (see for example p. 71 in Durrett [8]).

### 3.1.1 Mixture of Markov Models and Parameter Estimation

We adopt the theory behind the mixtures of Markov models from Melnykov [17]. Denote $\boldsymbol{Y}_i = (Y_{i1}, Y_{i2}, \ldots, Y_{iT_i}\}$ the $i$th categorical sequence of length $T_i$. Then, $Y_{i1}$ represents the first element in the $i$th sequence. Assuming the first-order Markov chain and denoting the joint probability mass function of $\boldsymbol{Y}_i$ by $p(\boldsymbol{y}_i) = \mathbb{P}[Y_{i1} = y_{i1}, Y_{i2} = y_{i2}, \ldots, Y_{iT_i} = y_{iT_i}]$, we obtain $p(\boldsymbol{y}_i) = \mathbb{P}[Y_{i1} = y_{i1}] \prod_{t=2}^{T_i} \mathbb{P}[Y_{it} = y_{it} | Y_{i(t-1)} = y_{i(t-1)}]$. Next, denote $\alpha_{y_i 1} = \mathbb{P}[Y_{i1} = y_{i1}]$ and $\gamma_{y_{i(t-1)} y_{it}} = \mathbb{P}[Y_{it} = y_{it} | Y_{i(t-1)} = y_{i(t-1)}]$. Then, each $\gamma_{y_{i(t-1)} y_{it}}$, in fact, represents an element of the probability transition matrix

$$\boldsymbol{\Gamma} = \begin{pmatrix} \gamma_{11} & \gamma_{12} & \cdots & \gamma_{1J} \\ \gamma_{21} & \gamma_{22} & \cdots & \gamma_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{J1} & \gamma_{J2} & \cdots & \gamma_{JJ} \end{pmatrix},$$

where $J$ is the number of states of the Markov chain. Furthermore, denoting the number of realized transitions from state $j$ to state $j'$ for the $i$th categorical sequence as $x_{ijj'}$, we can rewrite (1.2) as

$$f(y_{i1}, \boldsymbol{x}_i; \boldsymbol{\vartheta}) = \sum_{k=1}^{K} \pi_k \left( \prod_{j=1}^{J} \alpha_{kj}^{I[Y_{i1}=j]} \right) \left( \prod_{j=1}^{J} \prod_{j'=1}^{J} \gamma_{kjj'}^{x_{ijj'}} \right), \tag{3.2}$$

for independent categorical sequential observations $\{y_{i1}, \boldsymbol{x}_i\}_{i=1,\ldots,n}$, where $\boldsymbol{x}_i$ is a $J \times J$-dimensional matrix with elements $x_{ijj'}$. Index $k$ in parameters $\alpha_{kj}$ and $\gamma_{kjj'}$ indicates that they are associated with the $k$th mixture component and $I[\cdot]$ represents the indicator function.

Given the probability mass function (3.2), the corresponding likelihood function is given by

$$L(\boldsymbol{\vartheta} | \{y_{i1}, \boldsymbol{x}_i\}_{i=1,\ldots,n}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \left( \prod_{j=1}^{J} \alpha_{kj}^{I[Y_{i1}=j]} \right) \left( \prod_{j=1}^{J} \prod_{j'=1}^{J} \gamma_{kjj'}^{x_{ijj'}} \right).$$

The standard procedure for handling estimation in problems with missing data is the EM algorithm introduced in Section 1.2. If unavailable membership labels $z_1, \ldots, z_n$ of all observed sequences are assumed to be known, the complete-data likelihood function can be written as

$$L_C(\boldsymbol{\vartheta}|\{y_{i1},\boldsymbol{x}_i\}_{i=1,\dots,n}) = \prod_{i=1}^{n}\prod_{k=1}^{K}\left[\pi_k\left(\prod_{j=1}^{J}\alpha_{kj}^{I[Y_{i1}=j]}\right)\left(\prod_{j=1}^{J}\prod_{j'=1}^{J}\gamma_{kjj'}^{x_{ijj'}}\right)\right]^{I[Z_i=k]}.$$

Then, we can derive the Q-function which is the conditional expectation of the complete log-likelihood function given observed data and parameter estimates from the previous iteration $\boldsymbol{\vartheta}^{(b-1)}$. In the E-step of the EM algorithm, we need to calculate the posterior probabilities by

$$\pi_{ik} = \frac{\pi_k^{(b-1)}\alpha_{ky_{i1}}^{(b-1)}\prod_{j=1}^{J}\prod_{j'=1}^{J}(\gamma_{kjj'}^{(b-1)})^{x_{ijj'}}}{\sum_{k'=1}^{K}\pi_{k'}^{(b-1)}\alpha_{k'y_{i1}}^{(b-1)}\prod_{j'=1}^{J}(\gamma_{kjj'}^{(b-1)})^{x_{ijj'}}}.$$

By maximizing the Q-function subject to $\sum_{k=1}^{K}\pi_k = 1, \sum_{j=1}^{J}\alpha_{kj} = 1$, and $\sum_{j'=1}^{J}\gamma_{kjj'} = 1$ for every $k = 1,\dots,K$, and $j = 1,\dots,J$, we obtain formulas for updating the parameter estimates. Then, the M-step consists of updating the parameters by the following expressions

$$\pi_k^{(b)} = \frac{\sum_{i=1}^{n}\pi_{ik}^{(b)}}{n}, \quad \alpha_{kj}^{(b)} = \frac{\sum_{i=1}^{n}\pi_{ik}^{(b)}I[Y_{i1}=j]}{\sum_{i=1}^{n}\pi_{ik}^{(b)}}, \quad \gamma_{kjj'}^{(b)} = \frac{\sum_{i=1}^{n}\pi_{ik}^{(b)}x_{ijj'}}{\sum_{i=1}^{n}\pi_{ik}^{(b)}\sum_{r'=1}^{J}x_{ijr'}}.$$

Upon convergence, the EM algorithm yields the MLE $\widehat{\boldsymbol{\vartheta}}$, i.e., $\widehat{\pi}_k, \widehat{\alpha}_{kj}$, and $\widehat{\gamma}_{kjj'}$ for all $k, j$, and $j'$.

## 3.2 Simulation Study

Similarly to the simulation study with Gaussian mixtures in Section 2.2, the goal was to examine and compare the outcomes of EM, CEM, and SEM with respect to the quality of clusterings measured by the $\mathcal{AR}$ index. In addition, we compared the success rates of choosing the correct number of clusters for the three procedures.

To scrutinize the performance of the three procedures, we arranged a specific, rather challenging family of scenarios. We assumed a mixture of three Markov models with

$J = 4$ states with the following parameters:

$$\boldsymbol{\pi} = (1/3, 1/3, 1/3)^T,$$
$$\boldsymbol{\alpha}_1 = \boldsymbol{\alpha}_2 = \boldsymbol{\alpha}_3 = (0.25, 0.25, 0.25, 0.25),$$
$$\boldsymbol{\Gamma}_1 = \begin{pmatrix} 0.3 & 0.5 & 0.1 & 0.1 \\ 0.5 & 0.3 & 0.1 & 0.1 \\ 0.4 & 0.4 & 0.1 & 0.1 \\ 0.4 & 0.4 & 0.1 & 0.1 \end{pmatrix}, \quad \boldsymbol{\Gamma}_2 = \begin{pmatrix} 0.1 & 0.1 & 0.5 & 0.3 \\ 0.1 & 0.1 & 0.3 & 0.5 \\ 0.1 & 0.1 & 0.4 & 0.4 \\ 0.1 & 0.1 & 0.4 & 0.4 \end{pmatrix}, \quad \boldsymbol{\Gamma}_3 = \begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{pmatrix}.$$

We selected weights $\boldsymbol{\pi}$ and initial distributions $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha}_3$ to be evenly distributed because we wanted to focus on the ability of the procedures to distinguish between the three transition matrices $\boldsymbol{\Gamma}_1, \boldsymbol{\Gamma}_2, \boldsymbol{\Gamma}_3$ that were chosen such that $\boldsymbol{\Gamma}_1$ describes a Markov process that tends to stay in states 1 and 2. On the contrary, $\boldsymbol{\Gamma}_2$ corresponds to a process that is inclined to stick with states 3 and 4. To make it more challenging, we introduced some noise between those two opposite processes by assuming that the third mixing process follows an evenly distributed transition matrix $\boldsymbol{\Gamma}_3$.

Then, we generated data sets of different combinations of sample sizes $N$ (the number of categorical sequences in a data set) and the lengths of sequences $T$ (we assumed for simplicity that all sequences have the same length $T$). From the law of large numbers, it is natural that for data sets with large $N$ and $T$, the resulting fit will detect clusters that very accurately correspond to the mixing components from which the data originated. However, for data sets of small sample sizes or with short sequences, recognizing the clusters will be much more challenging. To capture that convergence process, we chose $T = 10, 12, 15, 17, 20, 50$ and $N = 99, 300, 600, 1500$.

For each $(N, T)$-combination, we generated 100 data sets and ran the standard EM, CEM, and SEM to fit a mixture model with the number of components $K$ ranging from 1 to 5 while assuming that the number of states $J$ is known to be 4.

As we mentioned in Section 1.2, EM procedures may run into an issue of being stuck in a local maximum. We addressed that issue by implementing random initialization of parameters. That improved the chances of finding the best solution, especially for the standard EM. In each run, we generated 50 random initializations and chose the best model in terms of the number of clusters $K$ based on the BIC. As the stopping criterion, we used the relative change between two subsequent values of the log-likelihood function and also we capped the number of maximum iterations to stop the CEM procedure to avoid falling into a loop when a small number of states is being changed cyclically which may sometimes occur.

To implement the random initialization, we had to account for the fact that an initial distribution $\boldsymbol{\alpha}_k$ is a probability vector as well as the rows of the transition matrix $\boldsymbol{\Gamma}_k$ are probability vectors. A probability vector is characteristic by having all its elements from the interval $[0, 1]$ and all its elements sum up to one. A way to generate a probability vector is to draw a sample from the Dirichlet distribution. The Dirichlet distribution, often denoted as $\mathrm{Dir}(\boldsymbol{\alpha})$, is a multivariate generalization of the Beta distribution [11] which means that all its marginal distributions are Beta distributions. For us, the important case is when $\boldsymbol{\alpha}$ is a $J$-dimensional vector of ones. Then, all the marginals of $\mathrm{Dir}(\boldsymbol{\alpha})$ are uniform distributions $U(0, 1)$, and the elements of such a Dirichlet variable $(U_1, \ldots, U_J)$ sum up to 1. Therefore, every observation from $\mathrm{Dir}(\boldsymbol{\alpha})$ is a probability vector.

The results of the simulation study are summarized in Table 3.1 and Table 3.2. As expected, the median $\mathcal{AR}$ increases with the growing sample size $N$ as well as with the increasing length of sequences $T$. Also, the number of procedures that ended up with a mixture with the correct number of components (we assumed $K = 3$) increases with $N$ and $T$. The accuracy increases further as the median $\mathcal{AR}$ approaches values close to 1 for scenarios with $T = 50$. Next, we can observe that no procedure identified the correct $K$ in any simulation for the lowest $N$ or $T$. On the contrary, for all scenarios with $T = 50$, the correct $K$ was identified in all simulations. Furthermore, notice that in the scenario with $N = 99$ and $T = 10$, the median $\mathcal{AR}$ is 0 for CEM and SEM. This is because in both cases the procedures identified only one cluster in more than half of the number of simulations. Since the $\mathcal{AR}$ was zero for more than half of the simulations, the median $\mathcal{AR}$ is then zero as well. In several scenarios, the interquartile range (IQR) of $\mathcal{AR}$ is relatively big compared to other scenarios. Such phenomenon arises in scenarios where the transitioning towards the correct number of components is happening, for example for $N = 300, T = 20$. This is because the difference between $\mathcal{AR}$ of a model with incorrect number $K$ and a model with correct $K$ is relatively big, thus if a significant number of simulations end up with both models with incorrect $K$ and the correct $K$ (or $K$ that is closer to the correct $K$), the variability of $\mathcal{AR}$ across all 100 models is larger which leads to wider IQR.

Comparing EM, CEM, and SEM based on the results shown in Table 3.1 and Table 3.2, we can conclude that the outcomes of those three procedures are remarkably similar when it comes to fitting a mixture of Markov chains. However, upon a closer inspection, the standard EM appears to be superior when it comes to accuracy. The standard EM has the largest median $\mathcal{AR}$ in almost every scenario and in the scenarios where it does not, the difference is so small that it may be considered a statistical error. Similarly, the IQRs of $\mathcal{AR}$ are usually the smallest for the standard EM. The standard EM also seem to recognize the correct number $K$ faster (for smaller $N$ and $T$). Nevertheless, if a rapid calculation is required, CEM or SEM can be used without too many concerns of loosing accuracy. However, while CEM and SEM may be faster for the first couple of steps, the procedures may provide slower progress in later stages in contrast to the standard EM where the improvement is guaranteed with each iteration. Thus, the best approach would probably be to run CEM or SEM for the first couple of steps and then run the standard EM to achieve the best speed and accuracy.

| | | T=10 | | | T=12 | | | T=15 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | EM | CEM | SEM | EM | CEM | SEM | EM | CEM | SEM |
| $N = 99$ | $\mathcal{AR}$ med | 0.3618 | 0.0000 | 0.0000 | 0.4120 | 0.3955 | 0.3945 | 0.4379 | 0.4379 | 0.4369 |
| | $\mathcal{AR}$ IQR | 0.1142 | 0.3878 | 0.3796 | 0.0533 | 0.0650 | 0.0592 | 0.0303 | 0.0315 | 0.0361 |
| | $K = 1$ | 16 | 58 | 52 | 0 | 10 | 8 | 0 | 0 | 0 |
| | $K = 2$ | 84 | 42 | 48 | 100 | 90 | 92 | 100 | 100 | 100 |
| | $K = 3$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 5$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $N = 300$ | $\mathcal{AR}$ med | 0.4048 | 0.4002 | 0.4001 | 0.4245 | 0.4230 | 0.4212 | 0.4344 | 0.4341 | 0.4338 |
| | $\mathcal{AR}$ IQR | 0.0333 | 0.0272 | 0.0275 | 0.0227 | 0.0242 | 0.0264 | 0.0150 | 0.0178 | 0.0180 |
| | $K = 1$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 2$ | 100 | 100 | 100 | 100 | 100 | 100 | 99 | 100 | 100 |
| | $K = 3$ | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| | $K = 4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 5$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $N = 600$ | $\mathcal{AR}$ med | 0.4082 | 0.4054 | 0.4054 | 0.4234 | 0.4216 | 0.4225 | 0.4349 | 0.4320 | 0.4341 |
| | $\mathcal{AR}$ IQR | 0.0222 | 0.0213 | 0.0228 | 0.0116 | 0.0094 | 0.0105 | 0.1344 | 0.0116 | 0.0106 |
| | $K = 1$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 2$ | 100 | 100 | 100 | 99 | 100 | 100 | 67 | 94 | 100 |
| | $K = 3$ | 0 | 0 | 0 | 1 | 0 | 0 | 33 | 6 | 0 |
| | $K = 4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 5$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $N = 1500$ | $\mathcal{AR}$ med | 0.4095 | 0.4072 | 0.4057 | 0.4310 | 0.4228 | 0.4221 | 0.6220 | 0.6030 | 0.5532 |
| | $\mathcal{AR}$ IQR | 0.0133 | 0.0113 | 0.0099 | 0.1063 | 0.0091 | 0.0082 | 0.0242 | 0.0368 | 0.0405 |
| | $K = 1$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 2$ | 99 | 100 | 100 | 49 | 100 | 100 | 0 | 12 | 16 |
| | $K = 3$ | 1 | 0 | 0 | 51 | 0 | 0 | 100 | 88 | 84 |
| | $K = 4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 5$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table 3.1: Results of the simulation study for comparing the outcomes of EM, CEM, and SEM based on the Adjusted Rand Index ($\mathcal{AR}$) and on the number of identified clusters $K$. For example, in the scenario with $N = 99$ and $T = 10$, the standard EM ended up preferring a model with only one component for 16/100 simulations, while 84/100 simulations ended up with a model with two components. The model selection is carried out based on minimal BIC and $\mathcal{AR}$ med stands for median $\mathcal{AR}$ and IQR for interquartile range. Results for higher $T$ are available in Table 3.2.

| | | T=17 | | | T=20 | | | T=50 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | EM | CEM | SEM | EM | CEM | SEM | EM | CEM | SEM |
| $N = 99$ | $\mathcal{AR}$ med | 0.4428 | 0.4389 | 0.4379 | 0.4468 | 0.4414 | 0.4468 | 0.9400 | 0.9396 | 0.9396 |
| | $\mathcal{AR}$ IQR | 0.0349 | 0.0435 | 0.0297 | 0.0275 | 0.0192 | 0.0196 | 0.0590 | 0.0590 | 0.0590 |
| | $K = 1$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 2$ | 100 | 100 | 100 | 99 | 100 | 100 | 0 | 0 | 0 |
| | $K = 3$ | 0 | 0 | 0 | 1 | 0 | 0 | 100 | 100 | 100 |
| | $K = 4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 5$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $N = 300$ | $\mathcal{AR}$ med | 0.4421 | 0.4401 | 0.4424 | 0.6757 | 0.4519 | 0.4475 | 0.9603 | 0.9602 | 0.9604 |
| | $\mathcal{AR}$ IQR | 0.0147 | 0.0098 | 0.0138 | 0.2867 | 0.2442 | 0.1883 | 0.0267 | 0.0267 | 0.0196 |
| | $K = 1$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 2$ | 92 | 98 | 99 | 26 | 67 | 72 | 0 | 0 | 0 |
| | $K = 3$ | 8 | 2 | 1 | 74 | 33 | 28 | 100 | 100 | 100 |
| | $K = 4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 5$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $N = 600$ | $\mathcal{AR}$ med | 0.6593 | 0.5972 | 0.4430 | 0.7217 | 0.7099 | 0.6914 | 0.9604 | 0.9555 | 0.9508 |
| | $\mathcal{AR}$ IQR | 0.0379 | 0.1970 | 0.1492 | 0.0370 | 0.0441 | 0.0454 | 0.0182 | 0.0148 | 0.0231 |
| | $K = 1$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 2$ | 7 | 42 | 73 | 0 | 0 | 1 | 0 | 0 | 0 |
| | $K = 3$ | 93 | 58 | 27 | 100 | 100 | 99 | 100 | 100 | 100 |
| | $K = 4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 5$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $N = 1500$ | $\mathcal{AR}$ med | 0.6706 | 0.6495 | 0.6193 | 0.7272 | 0.7138 | 0.6985 | 0.9565 | 0.9546 | 0.9546 |
| | $\mathcal{AR}$ IQR | 0.0274 | 0.0296 | 0.0274 | 0.0226 | 0.0333 | 0.0233 | 0.0112 | 0.0118 | 0.0126 |
| | $K = 1$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 2$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 3$ | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| | $K = 4$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $K = 5$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table 3.2: Second part of Table 3.1.

## 3.3 Modeling the U.S. Senate Elections by a Mixture of Markov Chains

We fit different mixtures of Markov models to a data set consisting of categorical data representing the results of the United States Senate elections. The EM algorithm is employed to estimate the model parameters, including the number of components $K$ and the order of the Markov model. Then, we use the fitted models for clustering the U.S. states which can provide us with some insight into what certain states have in common and we demonstrate how such a model can be used for forecasting by predicting the results of the next Senate elections in 2022.

### 3.3.1 U.S. Senate Elections

The foundations of the U.S. Senate were laid by the U.S. Constitution [1, Article I, Section 3]. Originally, senators were chosen by state legislatures. However, in 1913, the 17th Amendment provided that senators would be directly elected by the people.

From the beginning, it was established that each state would be represented by two senators who are elected to serve six-year terms. Moreover, senators were divided into three classes, so that the elections held every two years fill one third of the seats. Nowadays, there are 100 seats evenly distributed into Class I, Class II, and Class III and elections are held in November of every even-numbered year. In case a senator resigns or cannot finish their term, special elections can be held to elect a senator who will finish the incomplete term. Special elections are usually held together with general elections but some states, such as Alabama and Texas, allow for arrangement of special elections prior to general elections. Before the vacancy is filled by a newly elected senator, states legislatures can make temporarily appointments. Additional information about special elections in the U.S. can be found in [2].

### 3.3.2 Data

Our U.S. Senate elections data set consists of categorical sequential data organized into a table whose rows correspond to the U.S. states and the columns to the years when elections were held. The values of the table are strings consisting of "D" for a Democratic incumbent or "R" for a Republican incumbent.

The data set starts in 1966 when the first U.S. Senate elections were held after the ratification of the Civil Rights Act in 1964. The Civil Rights Act was an important milestone in the U.S. politics because it established, among others, the right to vote for African Americans. Therefore, not only the U.S. electorate changed but also the two main political parties in the U.S. adjusted their policies and ideological direction, which led to the evolution of modern Democrats and Republicans. The latest data come from 2020 Senate elections.

| State | 1966 | 1968 | 1970 | 1972 | 1974 | 1976 | 1978 | 1980 | 1982 | $\cdots$ | 2020 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Minnesota | D | NA | D | D | NA | D/R | R | NA | R | $\cdots$ | D |
| Mississippi | D | NA | D | D | NA | D | R | NA | D | $\cdots$ | R |
| Missouri | NA | D | D | NA | D | R | NA | D | R | $\cdots$ | NA |

Table 3.3: An excerpt from the U.S. Senate data set that illustrates the organization of the data.

In several cases during the reference period, an independent, or a candidate of the Conservative party was elected. In such cases, they were classified as "D" or "R" based on the party they caucused with. If a senator was elected in special elections to finish an unfinished term, the party memberships of the senators in such a term are separated by "/", for example "D/R" means that a Democratic incumbent did not finish their term and a Republican candidate was elected to fill the vacancy. If no elections were held for a given state in a given year, the value is "NA".
The data were collected from the official website of the U.S. Senate that provides lists of senators ordered by states, for example the data for Minnesota are available at [24]. The data set is stored in `US_Senate_election_data.xlsx` which can be accessed on Github [14]. We can see an excerpt from the data set in Table 3.3.

### 3.3.3 Mixture of Markov Models for the U.S. Senate Data Set

It appears plausible to model the data described in Subsection 3.3.2 by Markov models. For each U.S. state, we can derive a sequence whose values represent the incumbents elected subsequently. For example, consider the first row of Table 3.3. Ignoring the NA values, such a time series for Minnesota is $D, D, D, D, R, R, \ldots, D$ and it represents the party affiliation of every Minnesota senator elected from 1966.
Suppose that the probability of electing a Democrat in a given state in next elections depends only on the party affiliation of the current incumbent in that state. Naturally, we assume the same for a Republican candidate. Then, let $X_n^1$ be a 1st order Markov chain with state space $S = \{D, R\}$. Its transition matrix can be written as

$$\begin{pmatrix} P_{DD} & P_{DR} \\ P_{RD} & P_{RR} \end{pmatrix}, \tag{3.3}$$

where the matrix elements are transition probabilities. For example, $P_{DR}$ represents the probability that a republican candidate will be elected in the next election given that the current incumbent is Democratic.
However, the assumption of transition probabilities depending only on the current state might be too restrictive. For example, it is widely believed that an incumbent senator holds

an inherent advantage, especially if they were reelected more than once in the past. We can use a model with longer memory by constructing higher order Markov chains. Let $X_n^2$ be a second order Markov chain with state space $S = \{1, 2, 3, 4\}$, where state 1 represents the situation where two terms in a row were served by a Democratic senator ($DD$), state 2 represents $DR$, and similarly state 3 corresponds to $RD$ and state 4 to $RR$. Then, its transition matrix is

$$\begin{pmatrix} P_{11} & P_{12} & 0 & 0 \\ 0 & 0 & P_{23} & P_{24} \\ P_{31} & P_{32} & 0 & 0 \\ 0 & 0 & P_{43} & P_{44} \end{pmatrix}, \tag{3.4}$$

where for example $P_{23}$ is the probability that the Markov chain will make a transition from state 2 ($DR$) to state 3 ($RD$) which represents the probability that a Democratic candidate will be elected given that there is a Republican senator serving today and the senator before was Democratic. Notice that half of the transition probabilities of the second order Markov chain is zero because, for example, going from state $DR$, represented by 2, to states $DD$ and $DR$, represented by 1 and 2 respectively, is impossible.

Similarly, we can define a $p$th order Markov chain $X_n^p$ that will have $2^p$ states and will take $p - 1$ past states into account together with the current state. That being said, we will consider only models up to the third order, due to limitations imposed by the relatively short time-frame between 1966 and 2020.

Generally, we cannot assume that the population (in this case, the population is the outcome sequences for all U.S. states) is homogeneous. In fact, we expect it, in this case, to be heterogeneous which drives the motivation to use a mixture of Markov chains rather than just one chain to model the data and attempt to find clusters of U.S. states. For this reason, we consider a mixture of $K$ Markov chains that can be represented by (3.2). In this thesis, we consider only mixtures of Markov chains of the same order.

In the following sections we present our results from fitting the model to the Senate election data set.

### 3.3.4 Model Fitting and Selection

We fitted the Markov chain mixture model described in Section 3.1 (more specifically in Section 3.3.3) to the U.S. Senate election data set described in Subsection 3.3.2. Since we do not know what the true number of clusters and the order of Markov chains are, we need to estimate not only the vector of mixing proportions $\boldsymbol{\pi}$, the initial distributions of the states $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_K$, and the transition matrices $\boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_K$ (following the forms (3.3), (3.4), or a bigger matrix constructed in similar fashion, depending on the order of the Markov chains in the mixture), but also the number of components $K$ and the order of Markov chains.

| Order | 1st | 2nd | 3rd |
|---|---|---|---|
| K | BIC | BIC | BIC |
| 1 | 1284.6 | 1037.9 | 1218.7 |
| 2 | 1220.5 | 1080.8 | 1430.4 |
| 3 | 1228.8 | 1128.5 | |
| 4 | 1234.7 | | |
| 5 | 1242.5 | | |

Table 3.4: Comparison of mixture models with Markov chains of different orders and for different numbers of components $K$ based on BIC. We can see that the model with $K = 2$ provides the best fit among the first order Markov chain mixtures, while the model with $K = 1$ for the second and third-order chains. Overall, the best fit is achieved by a single second order Markov chain with $BIC = 1037.9$.

To account for that, we fitted the model for different combinations of $K$ and the orders of the Markov chains. To estimate the model parameters for a given order and $K$, we used the EM algorithm. We described the EM algorithm adapted to mixtures of Markov chains in Subsection 3.1.1. As we mentioned in Section 1.2, the likelihood function is generally not convex. To ensure that a solution obtained by the EM procedure is not only a local maximum, we conducted 100 random initializations of the EM algorithm for each combination. Since the model parameters are probability vectors (matrices), its elements must be from the interval $[0, 1]$ and they must add up to one across each vector (matrix row). To generate uniformly distributed $J$-dimensional probability vectors, we drew samples from the Dirichlet distribution $\text{Dir}(\boldsymbol{\alpha})$, with $\boldsymbol{\alpha}$ being a $J$-dimensional vector of ones. We proceeded in the same fashion as in the simulation study in Section 2.2.
Then we compared the fitted models. To avoid overfitting, we used BIC to compare the models, since BIC penalizes the number of parameters of a model unlike a comparison based merely on likelihood. In Table 3.4, we can see the comparison. Among the first order mixtures, the minimum BIC is attained for $K = 2$ but the values of BIC does not increase very quickly with adding more components. Also, it is clear that the first order Markov chain ($K = 1$) is too simple to model all the important characteristics of the data set. By far, the best fit is achieved by a single second order Markov chain. Since $K = 1$, it is not really a mixture but because it looks back one more step, it can capture more complex behavior using less parameters than mixtures of first-order chains. By adding more than one second order chain, BIC increases significantly. We also fitted a third order Markov chain and a mixture of two third-order models but their BIC's suggest that third order Markov chains are not a good model in this case.

### 3.3.5 Cluster Analysis

Although the models composed of first-order Markov chains provided generally suboptimal fit compared to the second-order Markov models, we still think that it is worth to present the parameter estimates and the clusters formed by those models and interpret them. To add some perspective to the study, Table 3.5 gives the estimates of the mixtures of first-order chains, while Figure 3.1 illustrates the resulting clusters tabulated in Table 4.1. The top left map depicts the outcome of the mixture with two components. These clusters can be interpreted as red states and blue states which is a common way to distinguish between states which vote for Republicans and states which favor Democrats [12]. Furthermore, the corresponding estimates of the transition matrices support this interpretation. The first matrix clearly corresponds to states that favor Democrats and the second matrix is almost a complete mirror image of the first matrix with the probabilities just a little bit more concentrated along the diagonal which suggests less frequent transitions between the two parties.

The top right map illustrates the clusters formed by the mixture with three components. In addition to two clusters that again resemble red and blue states, we have four states (Kansas, Idaho, Utah, Wyoming) that form a new cluster (depicted in dark red). The corresponding parameter estimates advocate that by increasing $K$ to 3, the cluster of red states was further separated into two groups that can be interpreted as red states and deep red states. Looking at the matrix corresponding to the dark red cluster, the estimated probability of the transition from R to D is 0.061 and the transition from D to R is estimated to happen with probability 1. Inspecting the election history of those four states, it can be observed that these states have a long and rarely, if ever, interrupted history of electing Republican senators in the given time frame. Then, the matrix corresponding to the red cluster suggests that after excluding the dark red states, the remaining states tend to be more open to transitioning between R and D.

Next map at the bottom left depicts the results of clustering by a four-component mixture. The corresponding matrix estimates suggest the presence of states with a strong preference for Democrats (blue), states that tend to vote for Democrats (purple), states that prefer Republicans (red), and states that maintain the status quo (grey), meaning that if a Democrat wins an election, a Democratic candidate will likely win again in the next election and vice versa. The grey cluster identifies states that consistently vote for one party (often for one senator who is being reelected again and again, e.g., an Alabama Senator Richard Shelby (R) was elected in 1986 and then reelected five times) and only rarely transitions between the two parties (in fact, some of those states transitioned only once, e.g., Arkansas, Louisiana, and Oregon which we discuss later).

The last map at the bottom right is a depiction of the clusters that were formed by a mixture of five Markov chains. Based on the transition matrix estimates, the only change compared to the four-component mixture is that a dark red cluster separated from the red cluster. Thus the five clusters can be interpreted as states with a strong preference for Democrats (blue), Democratic-leaning states (purple), Republican-leaning states (red), states with a strong preference for Republicans (dark red), and states that maintain the status quo (grey).

The reader might be surprised that in certain cases, some states were assigned to a cluster that goes against the conventional understanding of the current political situation. For example, the mixtures with $K = 2$ and $K = 3$, assigned Arkansas and Louisiana to the blue cluster, while those states are considered to be a part of the deep red South and on the contrary Oregon is in the red cluster, although it is considered to be a very liberal state nowadays. The reason for this is that we the data set we used starts in 1966 and, in fact, the mentioned states were not always voting red in case of Arkansas and Louisiana and voting blue in case of Oregon. The two South states were predominantly Democratic in the past. However, the Republican party has won every elections over there since 2010. Similarly, Oregon transitioned from a red state and Democrats won in every elections after 2002. Those examples show that political preferences can change relatively quickly, at least when it comes to the party affiliation of the elected candidates. The counterintuitive cluster assignment is likely caused by the the fact that the Markov models we use assume that transition probabilities are stationary in time. To improve the model so it captures the mentioned evolution of political preferences, transition probabilities could be assumed dependent on time. Thus, the transition matrix of the $k$th component $\mathbf{\Gamma}_k$ would become $\mathbf{\Gamma}_k(t)$.

### 3.3.6 Forecasting the 2022 Elections

As mentioned earlier, the best fit was achieved by a single second-order Markov model (Table 3.4). The mixtures of Markov models of orders higher than one had significantly higher BIC and provided no additional insight to this analysis. Below are the estimates of the transition probability matrix and the initial distribution of the states:

$$\widehat{\mathbf{\Gamma}} = \begin{pmatrix} 0.842 & 0.158 & 0 & 0 \\ 0 & 0 & 0.726 & 0.274 \\ 0.289 & 0.712 & 0 & 0 \\ 0 & 0 & 0.147 & 0.853 \end{pmatrix}, \quad \widehat{\boldsymbol{\alpha}} = (0.36, 0.08, 0.28, 0.28)^T.$$

Looking at the transition matrix that follows the form (3.4), we can conclude that the behavior of red and blue states is somehow covered entirely by this second order Markov model. For example, if a Democrat has served for two terms, the probability that a Democrat will be elected for the next term is 0.842. Vice versa, the probability that a Republican will be elected after two terms served by Republicans is 0.853. The advantage of this model compared to a mixture of two first-order chain is that it has much fewer parameters thus its BIC is lower. The following scheme lists all the possible outcomes of the fitted Markov model.

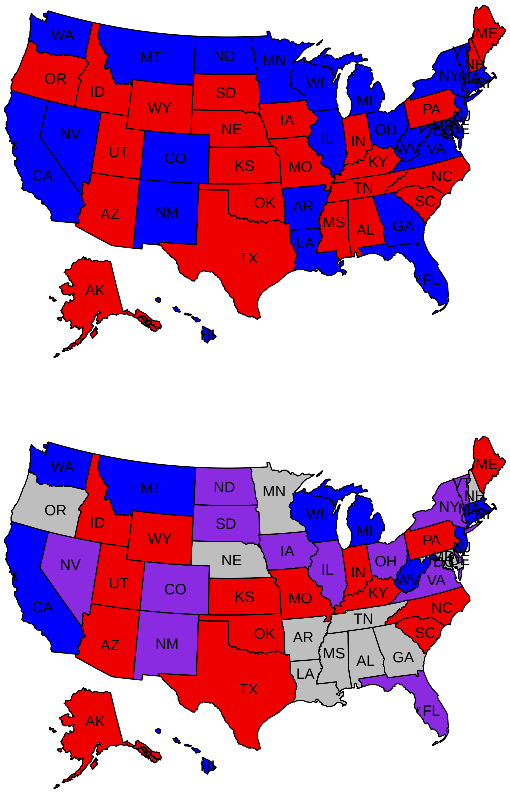
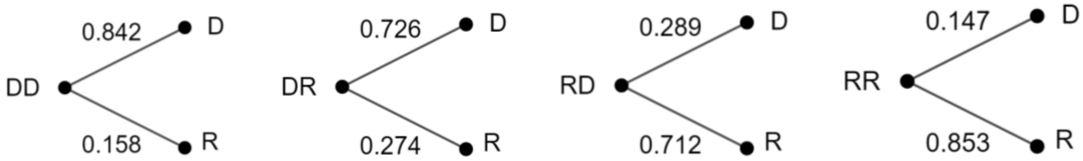


Figure 3.1: The U.S. States clusters formed by mixtures of first-order Markov models with $K = 2$ (top left), $K = 3$ (top right), $K = 4$ (bottom left), $K = 5$ (bottom right). The corresponding parameter estimates are available in Table 3.5.

| | Estimated parameters of mixtures of first-order Markov chains | | | | |
|---|---|---|---|---|---|
| $K = 2$ | $BIC = 1220.5$ | | | | |
| $\widehat{\boldsymbol{\pi}}$ | $(0.465, 0.535)^T$ | | | | |
| $\widehat{\boldsymbol{\alpha}}$ | $(0.541, 0.459)^T$, | $(0.324, 0.676)^T$ | | | |
| $\widehat{\boldsymbol{\Gamma}}$ | $\begin{pmatrix} 0.739 & 0.261 \\ 0.711 & 0.289 \end{pmatrix}$ | $\begin{pmatrix} 0.390 & 0.610 \\ 0.239 & 0.761 \end{pmatrix}$ | | | |
| Color | Blue | Red | | | |
| $K = 3$ | $BIC = 1227.8$ | | | | |
| $\widehat{\boldsymbol{\pi}}$ | $(0.497, 0.430, 0.073)^T$ | | | | |
| $\widehat{\boldsymbol{\alpha}}$ | $(0.550, 0.450)^T$, | $(0.388, 0.612)^T$, | $(0, 1)^T$ | | |
| $\widehat{\boldsymbol{\Gamma}}$ | $\begin{pmatrix} 0.748 & 0.252 \\ 0.734 & 0.266 \end{pmatrix}$, | $\begin{pmatrix} 0.423 & 0.577 \\ 0.300 & 0.700 \end{pmatrix}$, | $\begin{pmatrix} 0 & 1 \\ 0.061 & 0.939 \end{pmatrix}$ | | |
| Color | Blue | Red | Dark Red | | |
| $K = 4$ | $BIC = 1234.7$ | | | | |
| $\widehat{\boldsymbol{\pi}}$ | $(0.194, 0.286, 0.284, 0.236)^T$ | | | | |
| $\widehat{\boldsymbol{\alpha}}$ | $(0.714, 0.286)^T$, | $(0.290, 0.710)^T$, | $(0.27, 0.73)^T$, | $(0.600, 0.400)^T$ | |
| $\widehat{\boldsymbol{\Gamma}}$ | $\begin{pmatrix} 0.816 & 0.184 \\ 1 & 0 \end{pmatrix}$, | $\begin{pmatrix} 0.606 & 0.394 \\ 0.670 & 0.330 \end{pmatrix}$, | $\begin{pmatrix} 0.203 & 0.797 \\ 0.224 & 0.776 \end{pmatrix}$, | $\begin{pmatrix} 0.717 & 0.283 \\ 0.258 & 0.742 \end{pmatrix}$ | |
| Color | Blue | Purple | Red | Grey | |
| $K = 5$ | $BIC = 1242.5$ | | | | |
| $\widehat{\boldsymbol{\pi}}$ | $(0.190, 0.263, 0.218, 0.069, 0.260)^T$ | | | | |
| $\widehat{\boldsymbol{\alpha}}$ | $(0.713, 0.287)^T$, | $(0.298, 0.702)^T$, | $(0.341, 0.659)^T$, | $(0, 1)^T$, | $(0.585, 0.415)^T$ |
| $\widehat{\boldsymbol{\Gamma}}$ | $\begin{pmatrix} 0.818 & 0.182 \\ 1 & 0 \end{pmatrix}$, | $\begin{pmatrix} 0.618 & 0.382 \\ 0.698 & 0.302 \end{pmatrix}$, | $\begin{pmatrix} 0.206 & 0.794 \\ 0.305 & 0.695 \end{pmatrix}$, | $\begin{pmatrix} 0 & 1 \\ 0.054 & 0.946 \end{pmatrix}$, | $\begin{pmatrix} 0.705 & 0.295 \\ 0.272 & 0.728 \end{pmatrix}$ |
| Color | Blue | Purple | Red | Dark Red | Grey |

Table 3.5: The estimates of the parameters of first-order Markov chain mixtures for different number of components $K$. Each vector is comprised of two entries from which the first one corresponds to state $D$ and the second one to state $R$. All the matrices follow the form of matrix (3.3), i.e., a transition matrix whose first row and first column represent state $D$ and the second row and second column correspond to state $R$. The colors in the table correspond to colors used in Figure 3.1. The assigned clusters are tabulated in Table 4.1.



Such a model can be conveniently used to predict results of the elections in the future. Based on the knowledge of the outcomes of the elections in 2016, 2018, and 2020, we can forecast the results of 2022 Senate elections. The reason why we need the data from last three years for only a second-order Markov model is that elections are held twice every six years for each state thus data from only last two years would not be sufficient for some states. The forecast of 2022 Senate elections is summarized in Table 3.6. The model predicts that there will be elected 13 Democratic and 21 Republican candidates. Since the current distribution of the senators who will not finish their terms until 2024 is 36 Democrats to 30 Republicans, the Senate will be composed of 49 Democrats and 51 Republicans after the elections in 2022.

| State | 2016 | 2018 | 2020 | Input | 2022 |
|---|---|---|---|---|---|
| AL | R | NA | R | RR | R |
| AK | R | NA | R | RR | R |
| AZ | R/D | D | NA | DD | D |
| AR | R | NA | R | RR | R |
| CA | D | D | NA | DD | D |
| CO | D | NA | D | DD | D |
| CT | D | D | NA | DD | D |
| DE | NA | D | D | DD | NA |
| FL | R | R | NA | RR | R |
| GA | R | NA | D | RD | R |
| HI | D | D | NA | DD | D |
| ID | R | NA | R | RR | R |
| IL | D | NA | D | DD | D |
| IN | R | R | NA | RR | R |
| IA | R | NA | R | RR | R |
| KS | R | NA | R | RR | R |
| KY | R | NA | R | RR | R |
| LA | R | NA | R | RR | R |
| ME | NA | I(D) | R | DR | NA |
| MD | D | D | NA | DD | D |
| MA | NA | D | D | DD | NA |
| MI | NA | D | D | DD | NA |
| MN | NA | D | D | DD | NA |
| MS | NA | R | R | RR | NA |
| MO | R | R | NA | RR | R |
| MT | NA | D | R | DR | NA |
| NE | NA | R | R | RR | NA |
| NV | D | D | NA | DD | D |
| NH | D | NA | D | DD | D |
| NJ | NA | D | D | DD | NA |
| NM | NA | D | D | DD | NA |
| NY | D | D | NA | DD | D |
| NC | R | NA | R | RR | R |
| ND | R | R | NA | RR | R |
| OH | R | D | NA | RD | R |
| OK | R | NA | R | RR | R |
| OR | D | NA | D | DD | D |
| PA | R | D | NA | RD | R |
| RI | NA | D | D | DD | NA |
| SC | R | NA | R | RR | R |
| SD | R | NA | R | RR | R |
| TN | NA | R | R | RR | NA |
| TX | NA | R | R | RR | NA |
| UT | R | R | NA | RR | R |
| VT | I(D) | D | NA | DD | D |
| VA | NA | D | D | DD | NA |
| WA | D | D | NA | DD | D |
| WV | NA | D | R | DR | NA |
| WI | R | D | NA | RD | R |
| WY | NA | R | R | RR | NA |

Table 3.6: The table is comprised of historical data from the last three Senate elections, the input for the second-order Markov model, and the resulting forecast for 2022 Senate elections. The possible values of the historical data are explained in Subsection 3.3.2. The values in the column `Input` simply consist of the political affiliation of the two most recently elected senators. Finally, the column 2022 consists of the forecasted party affiliation of newly elected senators.

# Chapter 4

# Conclusion

First, we introduced the concept of finite mixture models, put them into historical perspective, and explained the principle of using mixture models as a model-based clustering framework. Then, we described the EM algorithm as a traditional method of the estimation of parameters of mixture models and based on the pitfalls that may accompany the EM procedure that we outlined, we introduced two variants of the EM algorithm, namely the classification EM (CEM) and the stochastic EM (SEM). The final part of the introduction covers model selection based on the Bayes Information criterion (BIC) and comparison of model partitionings (clusterings) using the Adjusted Rand index ($\mathcal{AR}$).

To have full control over the procedures, we implemented all the methods we used from the ground up without employing any specialized packages for mixture models. The only exception was the generator of synthetic data following a Gaussian mixture model for which we used R package `MixSim` [19]. All the codes that used in this project are available on Github [14].

In the second chapter, we introduced a finite Gaussian mixture and briefly described the derivation of the iterative formulas for the EM algorithm. Then, we discussed the specifics and common pitfalls of the procedure and their remedies while describing the simulation scenarios. Finally, we presented the results of the simulation study which focused on comparing the standard EM, CEM, and SEM based on the achieved accuracy of clustering measured by $\mathcal{AR}$. We concluded that the accuracy of the procedures decreases significantly with higher average pairwise overlap of the synthetic data set $\Omega$ but except for CEM, the procedures appeared to be stable when the number of dimensions of the data were increasing. Based on the experience that CEM and SEM are much faster than EM, at least at the beginning where the outlines or clusters are formed, we suggested to run CEM or SEM for the first couple of steps and then employ the standard EM to achieve higher accuracy while improving the speed and avoiding issues with convergence that may occur when CEM and SEM is used in later stages.

The third chapter briefly introduced Markov chains and their higher-order generalizations. Then, we described a mixture of Markov models and outlined the derivation of the formulas for the EM algorithm. In the practical part, we presented the results of the simulation study we conducted to examine and compare the accuracy of the standard EM, CEM, and SEM based on synthetic data in different scenarios. Similarly, as for the Gaussian mixture, we measured the accuracy of obtained clusterings, but in addition, we studied the ability of the procedures to identify the correct number of clusters. The three procedures achieved very similar results in all scenarios. However, a closer look revealed that the standard EM is superior to the other two procedures when it comes to both accuracy measured by median $\mathcal{AR}$, variability measured by the interquartile range of $\mathcal{AR}$, and the ability to identify the correct number of clusters. Also, SEM outperformed CEM in almost every scenario. Considering that CEM and SEM still have an advantage of faster progress for the first couple of steps, we concluded that it seems advisable to run CEM or SEM first to quickly outline the clusters and then run the standard EM which should lead to higher accuracy while avoiding convergence issues of CEM and SEM.

The second part of the third chapter was concerned with the study of the U.S. Senate elections using a mixture of Markov models. First, we covered the basic information about the election system in the U.S. and described the data set we had collected. Then, we specified the mixture model in the perspective of the data set and the topic at hand. Employing the EM algorithm, we estimated the model parameters including the number of components and the order of Markov models in the mixture. Based on the BIC, the best model turned out to be a single second-order Markov model. Unfortunately, such model cannot be used for clustering. Nevertheless, we presented the results of clusterings provided by mixtures of Markov chains as they appeared to provide insight. Based on the visual representation of the clusters and the corresponding parameter estimates, we identified clusters that resemble the common partition to red and blue states. Furthermore, we identified clusters representing the states with a very strong preference for the Republican party or states that maintain the status quo. The most complex model with five components identified clusters that can be interpreted as Democratic, Democratic-leaning, Republican-leaning, Republican, and the states maintaining the status quo. Finally, we demonstrated how the fitted model with the smallest BIC can be used to forecast outcomes of the future elections. The model predicted that in the 2022 elections, 13 Democratic and 21 Republican candidates will be elected which would lead to a total of 49 Democrats and 51 Republicans in the Senate after the 2022 election.

We also identified several examples where the models did not cluster the U.S. states in a plausible way. We explained that some states might have been clustered counterituitively because there was a major change in the election outcomes of those states in the past. For example, although Oregon is considered to be a solid blue state nowadays, it was clustered together with red states by our model because in the past, Republicans dominated the elections. We believe that this issue is caused mainly by the assumption of stationary transition probabilities in our model that is not satisfied in reality. We believe that modeling transition probabilities as functions of time will significantly improve the model.

# Appendix

| Order | 1st | | | | 2nd | | 3rd | Order | 1st | | | | 2nd | | 3rd |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| K | 2 | 3 | 4 | 5 | 2 | 3 | 2 | K | 2 | 3 | 4 | 5 | 2 | 3 | 2 |
| State | ID | ID | ID | ID | ID | ID | ID | State | ID | ID | ID | ID | ID | ID | ID |
| AL | 2 | 3 | 4 | 2 | 1 | 3 | 1 | MT | 1 | 2 | 1 | 1 | 1 | 3 | 1 |
| AK | 2 | 1 | 2 | 5 | 2 | 3 | 2 | NE | 2 | 3 | 4 | 2 | 1 | 3 | 1 |
| AZ | 2 | 1 | 2 | 5 | 1 | 3 | 1 | NV | 1 | 2 | 3 | 4 | 1 | 3 | 2 |
| AR | 1 | 3 | 4 | 2 | 1 | 3 | 1 | NH | 2 | 3 | 4 | 2 | 1 | 3 | 1 |
| CA | 1 | 2 | 1 | 1 | 1 | 3 | 1 | NJ | 1 | 2 | 1 | 1 | 1 | 1 | 1 |
| CO | 1 | 2 | 3 | 4 | 1 | 3 | 2 | NM | 1 | 2 | 3 | 4 | 1 | 3 | 1 |
| CT | 1 | 2 | 1 | 1 | 1 | 2 | 1 | NY | 1 | 2 | 3 | 4 | 1 | 2 | 1 |
| DE | 1 | 2 | 3 | 4 | 1 | 2 | 1 | NC | 2 | 1 | 2 | 5 | 1 | 3 | 2 |
| FL | 1 | 2 | 3 | 4 | 2 | 3 | 2 | ND | 1 | 2 | 3 | 4 | 1 | 1 | 1 |
| GA | 2 | 3 | 4 | 2 | 1 | 3 | 2 | OH | 1 | 3 | 3 | 4 | 1 | 3 | 1 |
| HI | 1 | 2 | 1 | 1 | 1 | 2 | 1 | OK | 2 | 1 | 2 | 5 | 1 | 3 | 1 |
| ID | 2 | 1 | 2 | 3 | 2 | 1 | 2 | OR | 2 | 3 | 4 | 2 | 1 | 3 | 1 |
| IL | 1 | 2 | 3 | 4 | 1 | 3 | 1 | PA | 2 | 1 | 2 | 5 | 1 | 3 | 1 |
| IN | 2 | 1 | 2 | 5 | 1 | 3 | 2 | RI | 1 | 2 | 1 | 1 | 1 | 3 | 1 |
| IA | 2 | 1 | 3 | 5 | 1 | 3 | 2 | SC | 2 | 1 | 2 | 5 | 2 | 1 | 2 |
| KS | 2 | 1 | 2 | 3 | 2 | 1 | 2 | SD | 2 | 3 | 3 | 2 | 1 | 3 | 2 |
| KY | 2 | 1 | 4 | 2 | 1 | 3 | 2 | TN | 2 | 3 | 4 | 2 | 1 | 3 | 1 |
| LA | 1 | 3 | 4 | 2 | 1 | 3 | 1 | TX | 2 | 1 | 2 | 5 | 2 | 1 | 2 |
| ME | 2 | 1 | 2 | 5 | 2 | 1 | 2 | UT | 2 | 1 | 2 | 3 | 2 | 1 | 2 |
| MD | 1 | 2 | 4 | 2 | 1 | 2 | 1 | VT | 1 | 2 | 3 | 4 | 1 | 2 | 1 |
| MA | 1 | 2 | 1 | 1 | 1 | 1 | 1 | VA | 2 | 3 | 4 | 2 | 1 | 3 | 1 |
| MI | 1 | 2 | 1 | 1 | 1 | 1 | 1 | WA | 1 | 2 | 1 | 1 | 1 | 3 | 1 |
| MN | 1 | 3 | 4 | 2 | 1 | 3 | 1 | WV | 1 | 2 | 1 | 1 | 1 | 3 | 1 |
| MS | 2 | 3 | 4 | 2 | 1 | 3 | 2 | WI | 1 | 2 | 1 | 1 | 1 | 3 | 1 |
| MO | 2 | 1 | 2 | 5 | 1 | 3 | 2 | WY | 2 | 1 | 2 | 3 | 2 | 1 | 2 |

Table 4.1: The U.S. states clustered by mixtures of $K$ Markov models of order one, two, or three. The cluster membership is represented by ID taking on values from $\{1, \ldots, K\}$ for a given $K$.

# Bibliography

[1] Constitution of the United States of America, 1787. URL https://www.refworld.org/docid/3ae6b54d1c.html. Accessed: 2021-02-25.

[2] Vacancies in the United States Senate. https://www.ncsl.org/research/elections-and-campaigns/vacancies-in-the-united-states-senate637302453.aspx, 2021. Accessed: 2021-02-25.

[3] J. D. Banfield and A. E. Raftery. Model-based gaussian and non-gaussian clustering. *Biometrics*, 49(3):803–821, 1993. ISSN 0006341X, 15410420. URL http://www.jstor.org/stable/2532201.

[4] C. Bouveyron and C. Brunet-Saumard. Model-based clustering of high-dimensional data: A review. *Computational Statistics & Data Analysis*, 71:52–78, 2014.

[5] G. Celeux and J. Diebolt. The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Computational Statistics Quarterly*, 2:73–82, 1985.

[6] G. Celeux and G. Govaert. A classification em algorithm for clustering and two stochastic versions. *Computational Statistics & Data Analysis*, 14(3):315–332, 1992. ISSN 0167-9473. DOI https://doi.org/10.1016/0167-9473(92)90042-E. URL https://www.sciencedirect.com/science/article/pii/016794739290042E.

[7] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. ISSN 00359246. URL http://www.jstor.org/stable/2984875.

[8] R. Durrett. *Essentials of Stochastic Processes*. Springer-Verlag New York, 2012. ISBN 978-1-4614-3615-7.

[9] L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985. URL https://EconPapers.repec.org/RePEc:spr:jclass:v:2:y:1985:i:1:p:193-218.

[10] E. Çinlar. *Introduction to stochastic processes*. Prentice-Hall, Englewood Cliffs, NJ, [nachdr.] edition, 1975. ISBN 0134980891. URL http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+021423008&sourceid=fbw_bibsonomy.

[11] S. Kotz, N. Balakrishnan, and N. Johnson. *Continuous Multivariate Distributions, Volume 1: Models and Applications*. Continuous Multivariate Distributions. Wiley, 2004. ISBN 9780471654032. URL https://books.google.com/books?id=EbPBXJ-N-m4C.

[12] M. S. LEVENDUSKY and J. C. POPE. Red states vs. blue states: Going beyond the mean. *The Public Opinion Quarterly*, 75(2):227–248, 2011. ISSN 0033362X, 15375331. URL http://www.jstor.org/stable/41288382.

[13] J. Li and H. Zha. Two-way poisson mixture models for simultaneous document classification and word clustering. *Computational Statistics and Data Analysis*, 50(1): 163–180, 2006. ISSN 0167-9473. DOI https://doi.org/10.1016/j.csda.2004.07.013. URL https://www.sciencedirect.com/science/article/pii/S0167947304002336. 2nd Special issue on Matrix Computations and Statistics.

[14] J. Matas. Finite mixture models capstone project codes, 2021. URL https://github.com/jantas/US_Senate_Election_Markov_Chain/.

[15] G. Mclachlan, D. Peel, K. Basford, and P. Adams. The emmix software for the fitting of mixtures of normal and t-components. *Journal of Statistical Software*, 4, 08 1999.

[16] G. J. McLachlan and D. Peel. *Finite mixture models / Geoffrey McLachlan, David Peel*. Wiley New York ; Chichester, 2000. ISBN 0471006262. URL https://nla.gov.au/nla.cat-vn764646.

[17] V. Melnykov. Model-based biclustering of clickstream data. *Computational Statistics & Data Analysis*, 93(C):31–45, 2016. DOI 10.1016/j.csda.2014.09.01. URL https://ideas.repec.org/a/eee/csdana/v93y2016icp31-45.html.

[18] V. Melnykov and R. Maitra. Finite mixture models and model-based clustering. *Statistics Surveys*, 4(none):80 – 116, 2010. DOI 10.1214/09-SS053. URL https://doi.org/10.1214/09-SS053.

[19] V. Melnykov, W.-C. Chen, and R. Maitra. MixSim: An R package for simulating data to study performance of clustering algorithms. *Journal of Statistical Software*, 51(12): 1–25, 2012. URL https://www.jstatsoft.org/v51/i12/.

[20] S. Newcomb. A generalized theory of the combination of observations so as to obtain the best result. *American Journal of Mathematics*, 8(4):343–366, 1886. ISSN 00029327, 10806377. URL http://www.jstor.org/stable/2369392.

[21] K. Pearson. Contributions to the Mathematical Theory of Evolution. *Philosophical Transactions of the Royal Society of London Series A*, 185:71–110, Jan. 1894. DOI 10.1098/rsta.1894.0003.

[22] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971. DOI 10.1080/01621459.1971.10482356. URL `https://www.tandfonline.com/doi/abs/10.1080/01621459.1971.10482356`.

[23] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2): 461–464, 1978. ISSN 00905364. URL `http://www.jstor.org/stable/2958889`.

[24] U. S. Senate. List of Minnesota Senators, 2021. URL `https://www.senate.gov/states/MN/senators.htm`. Accessed: 2021-02-25.

[25] Y. Vardi, L. A. Shepp, and L. Kaufman. A statistical model for positron emission tomography. *Journal of the American Statistical Association*, 80(389):8–20, 1985. ISSN 01621459. URL `http://www.jstor.org/stable/2288030`.

[26] J. H. Wolfe. Pattern clustering by multivariate mixture analysis. *Multivariate Behavioral Research*, 5:329–350, 1970.