BayesCVI: A Bayesian Cluster Validity Index

Nathakhun Wiroonsri

Department of Mathematics

King Mongkut's University of Technology Thonburi, Bangkok, Thailand

useR!2024, Salzburg, Austria

Joint work with O. Preedasawakul.

A bit about myself

Nathakhun Wiroonsri

Department of Mathematics, King Mongkut's University of Technology Thonburi

- ❑ B.Sc. in Mathematics
  Chulalongkorn University
- ❑ Master of Financial Mathematics
  North Carolina State University
- ❑ Ph.D. in Applied Mathematics
  University of Southern California

Research Interests:

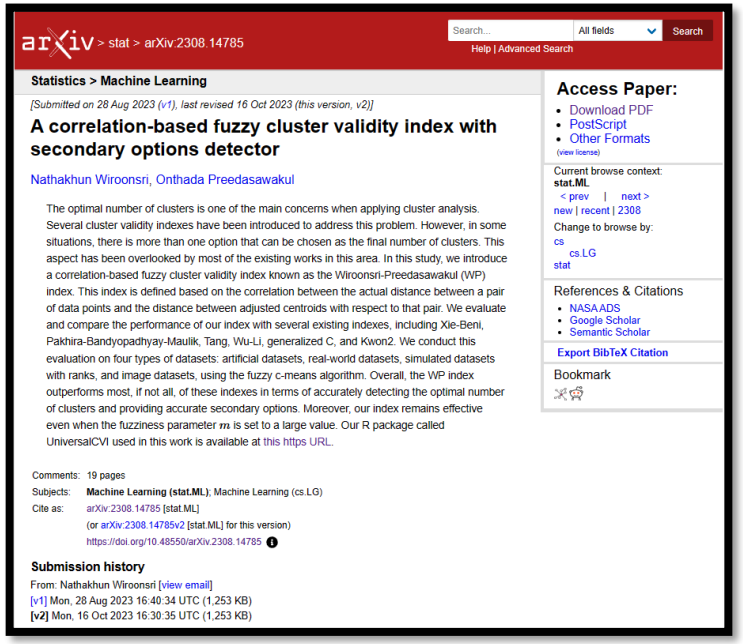
- Probability
- Distributional Approximation via Stein's Method
- Mathematical Statistics
- Statistical and Machine Learning

https://sites.google.com/view/nwiroonsri/

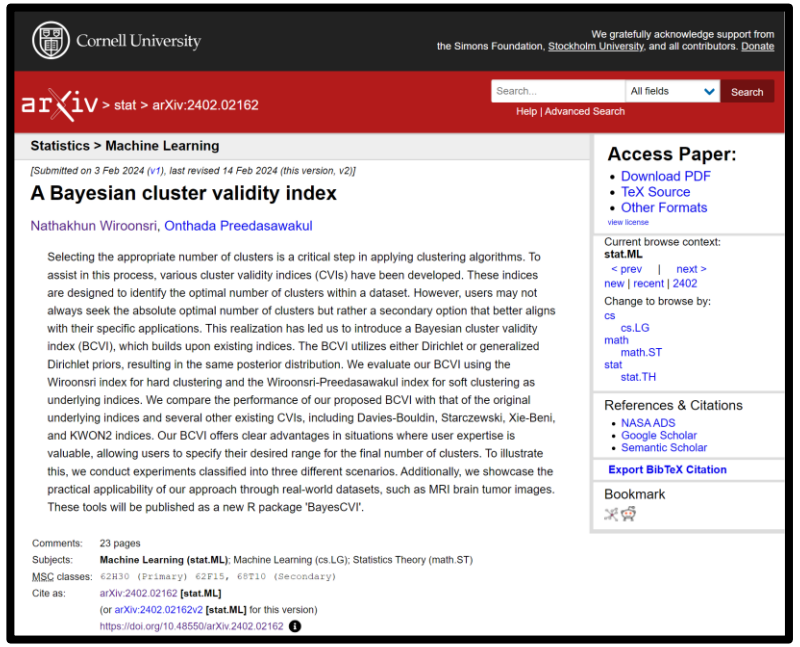
useR! Salzburg 2024

My talk will be based on:

W 2024



W and Preedasawakul 2024?



W and Preedasawakul 2024?

Supported by National Research Council of Thailand (NRCT), Grant number: N42A660991 (2023)

Outline

- Background
- Motivation
- Bayesian CVI
- R packages: BayesCVI, UniversalCVI
- Q&A

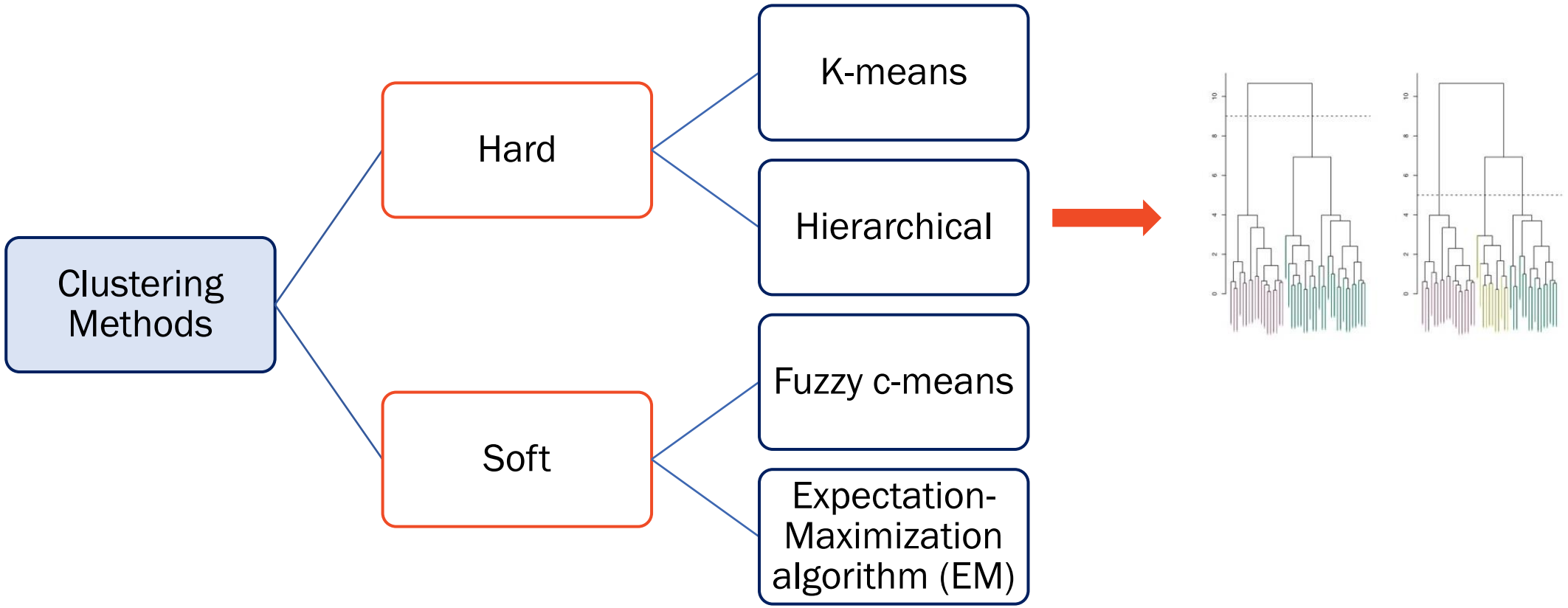
Cluster Analysis

Cluster analysis is an unsupervised learning tool used for grouping a set of objects so that objects in the same group share more similar characteristic than those in other groups.

Applications:

Image processing, pattern recognition, marketing, bioinformatics, social science, etc.

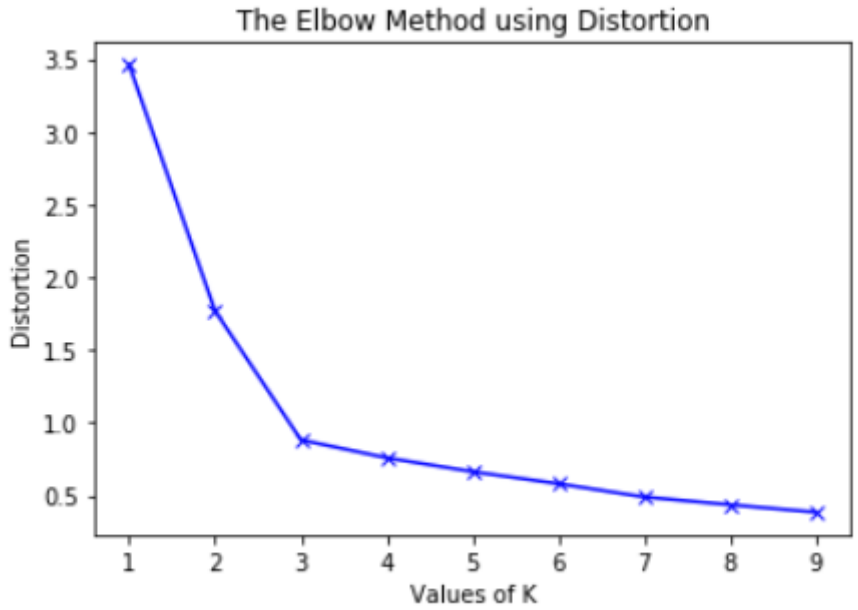
Variety of clustering methods



James et al. (2021)

Determine the number of clusters

- We have to specify the number of clusters. The classic method is called **Elbow method.**



https://www.geeksforgeeks.org/elbow-method-for-optimal-value-of-k-in-kmeans/

History of cluster validity indices

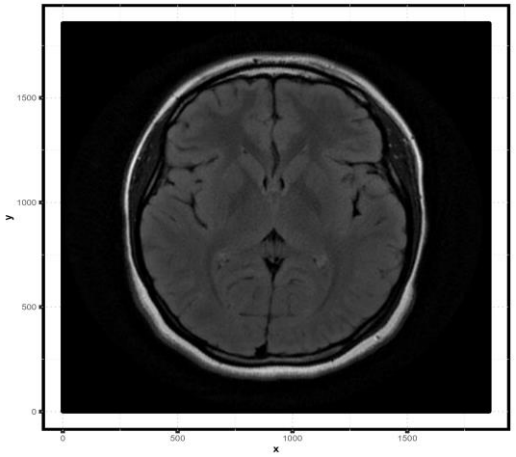
Hard clustering
Dunn's Index 1973
Calinski-Harabasz 1974
Davies-Bouldin's index 1979
Point biserial correlation 1980
Silhouette coefficient (Rousseeuw [1987], Sarle [1991])
Generalized Dunn index 1998
PBM index 2004
Chou-Su-Lai index 2004
Davies-Bouldin* index 2005
STR index 2017
Wiroonsri index 2021

Soft clustering
Xie–Beni (XB) index 1991
Pakhira–Bandyopadhyay–Maulik (PBM) index 2004
TANG index 2005
Wu–Li (WL) index 2015
Generalized C index 2016
KWON2 index 2021
Wiroonsri-Preedasawakul (WP) index 2023?

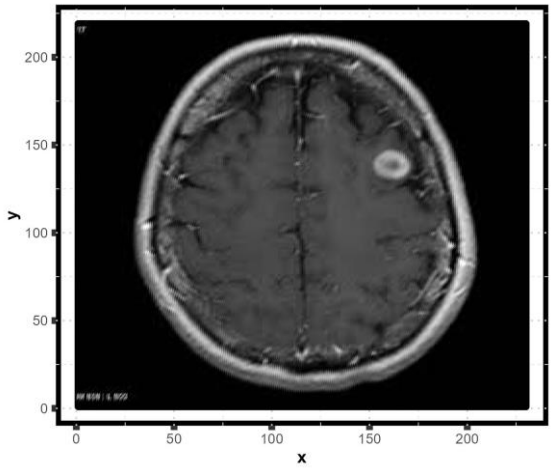
Motivation

What if the optimal number is not what we are looking for?

MRI: Brian Tumor Detection

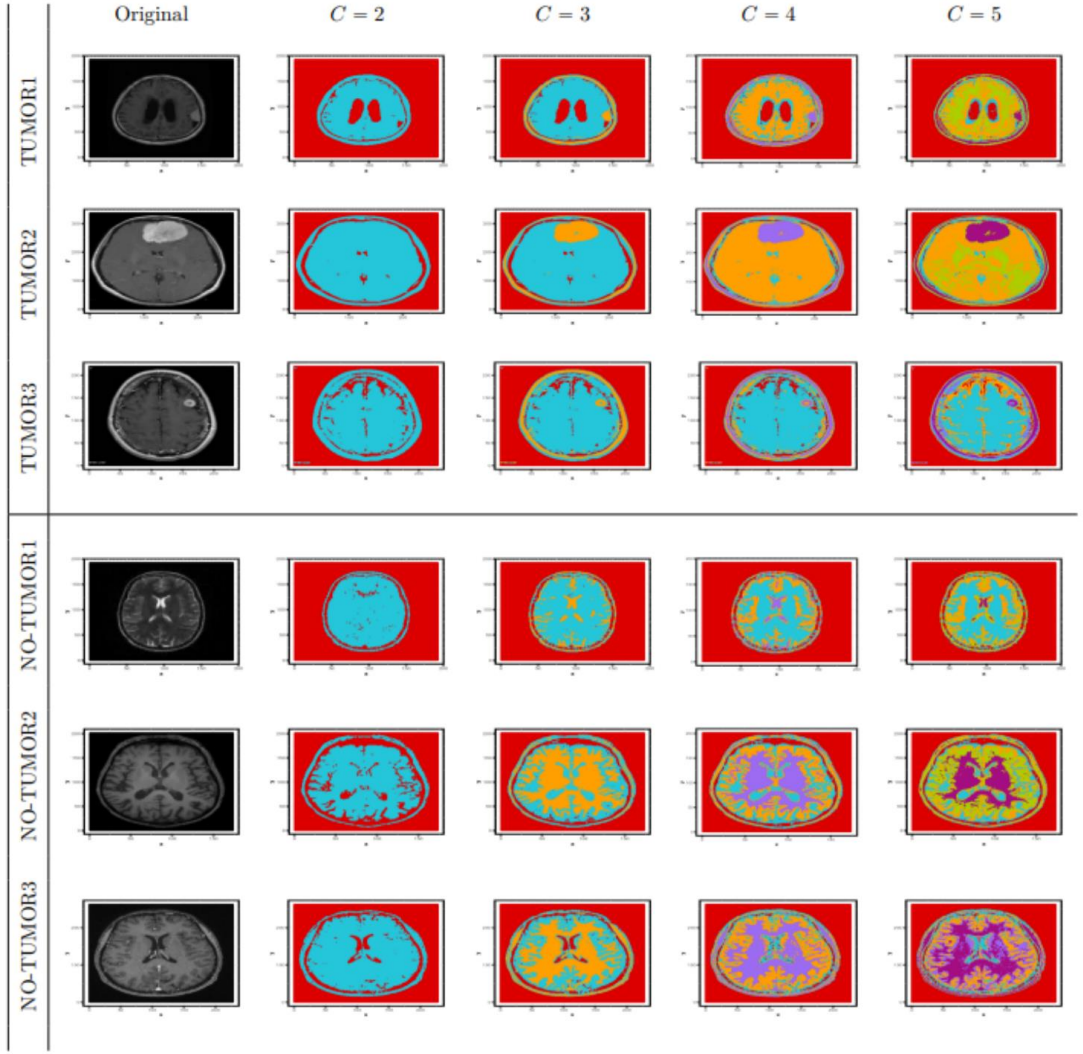


w/o tumor

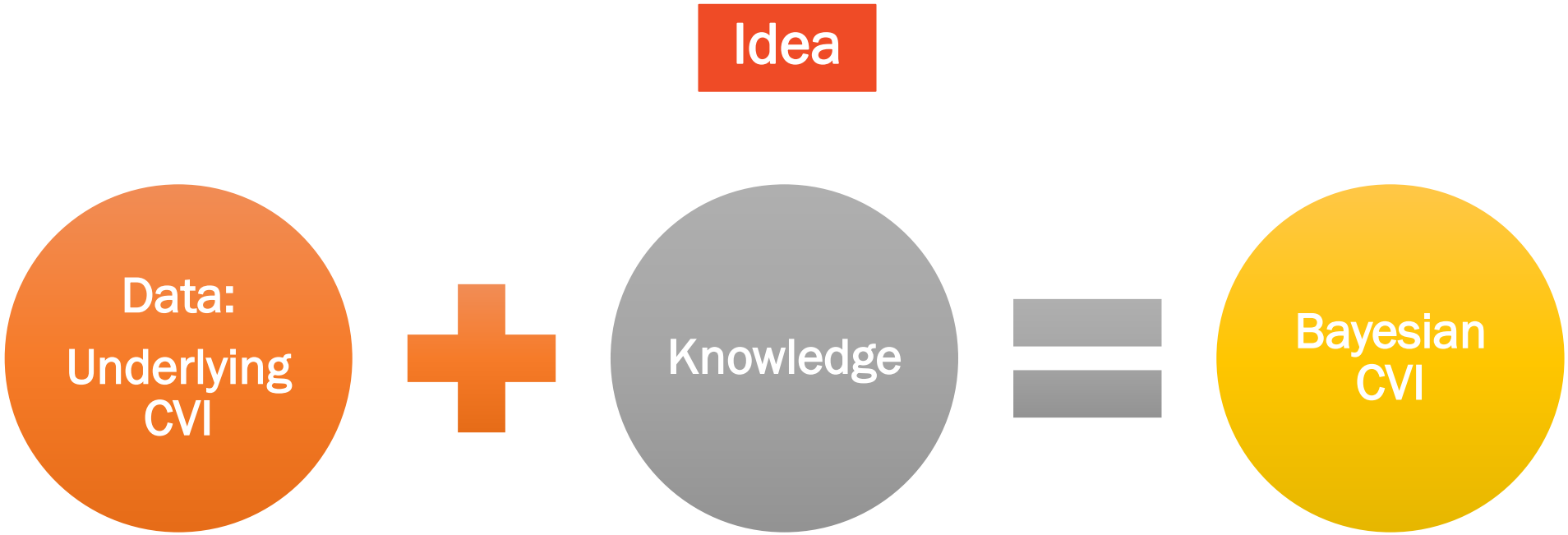


with tumor

https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection



Bayesian analysis and cluster validity index



Our indices

WI2021

Let $m \in \{2,3,\dots,n-1\}$ *and* $k = 2,3,\dots,m$

Case 1: $\max_{2 \le l \le m} NCI1(k) < +\infty$

$$\mathrm{NCI}_m(k) = \begin{cases} \min_{2 \leqslant l \leqslant m} \{\mathrm{NCI1}(l) | \mathrm{NCI1}(l) > -\infty\} & \text{if } \mathrm{NCI1}(k) = -\infty \\ \mathrm{NCI1}(k) & \text{otherwise,} \end{cases}$$

Case 2: $\max_{2 \le l \le m} NCI1(k) = +\infty$

$$\mathrm{NCI}_m(k) = \begin{cases} \min_{2 \leqslant l \leqslant m} \{\mathrm{NCI1}(l) | \mathrm{NCI1}(l) > -\infty\} + \mathrm{NCI2}(k) & \text{if } \mathrm{NCI1}(k) = -\infty \\ \max_{2 \leqslant l \leqslant m} \{\mathrm{NCI1}(l) | \mathrm{NCI1}(l) < +\infty\} + \mathrm{NCI2}(k) & \text{if } \mathrm{NCI1}(k) = +\infty \\ \mathrm{NCI1}(k) + \mathrm{NCI2}(k) & \text{otherwise,} \end{cases}$$

WPI2023

Let $p \in \{2,3,\dots,n-1\}$ *and* $k = 2,3,\dots,p$

Case1: $\max_{2 \le l \le p} \mathrm{WPCI1(k)} < +\infty$ and $\exists l \in [p] \backslash \{1\}$ such that $|\mathrm{WPCI1(l)}| < \infty$

$$\mathrm{WP}_p(k) = \begin{cases} \min_{2 \leqslant l \leqslant p} \{\mathrm{WPCI1}(l) | \mathrm{WPCI1}(l) > -\infty\} & \text{if } \mathrm{WPCI1}(k) = -\infty \\ \mathrm{WPCI1}(k) & \text{otherwise.} \end{cases}$$

Case 2: $\max_{2 \le l \le p} \mathrm{WPCI1(k)} = +\infty$ and $\exists\, l \in \{2,3,\dots,p\}$ such that $|\mathrm{WPCI1(l)}| < \infty$

$$\mathrm{WP}_p(k) = \begin{cases} \min_{2 \leqslant l \leqslant p} \{\mathrm{WPCI1}(l) | \mathrm{WPCI1}(l) > -\infty\} + \mathrm{WPCI2}(k) & \text{if } \mathrm{WPCI1}(k) = -\infty \\ \max_{2 \leqslant l \leqslant p} \{\mathrm{WPCI1}(l) | \mathrm{WPCI1}(l) < +\infty\} + \mathrm{WPCI2}(k) & \text{if } \mathrm{WPCI1}(k) = +\infty \\ \mathrm{WPCI1}(k) + \mathrm{WPCI2}(k) & \text{otherwise.} \end{cases}$$

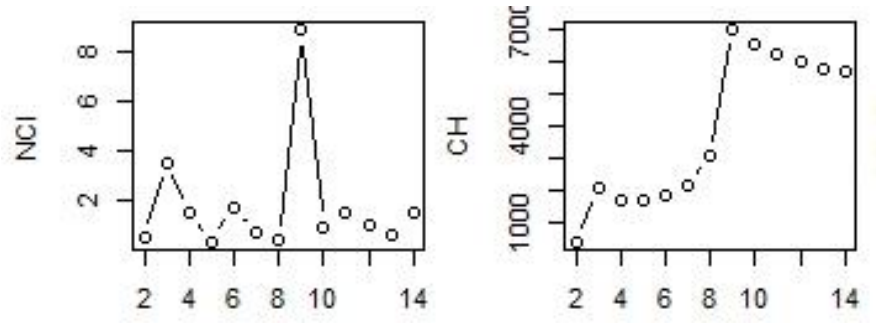
Case3 : $\forall l \in \{2,3,\dots,p\}, |\mathrm{WPCI1(l)}| = +\infty.$

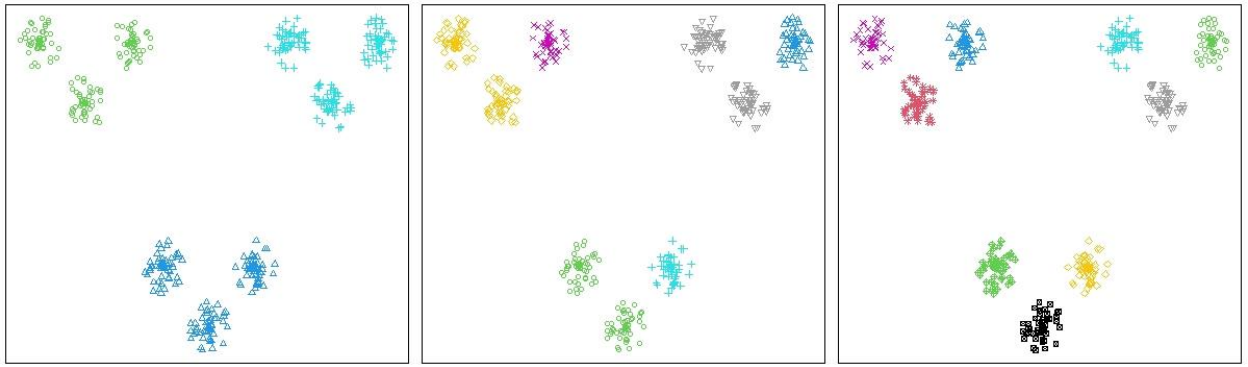
$$\mathrm{WP}_p(k) = \mathrm{WPCI2}(k),$$

Underlying CVIs Highlighted Features

- Very accurate in detecting the optimal number of clusters

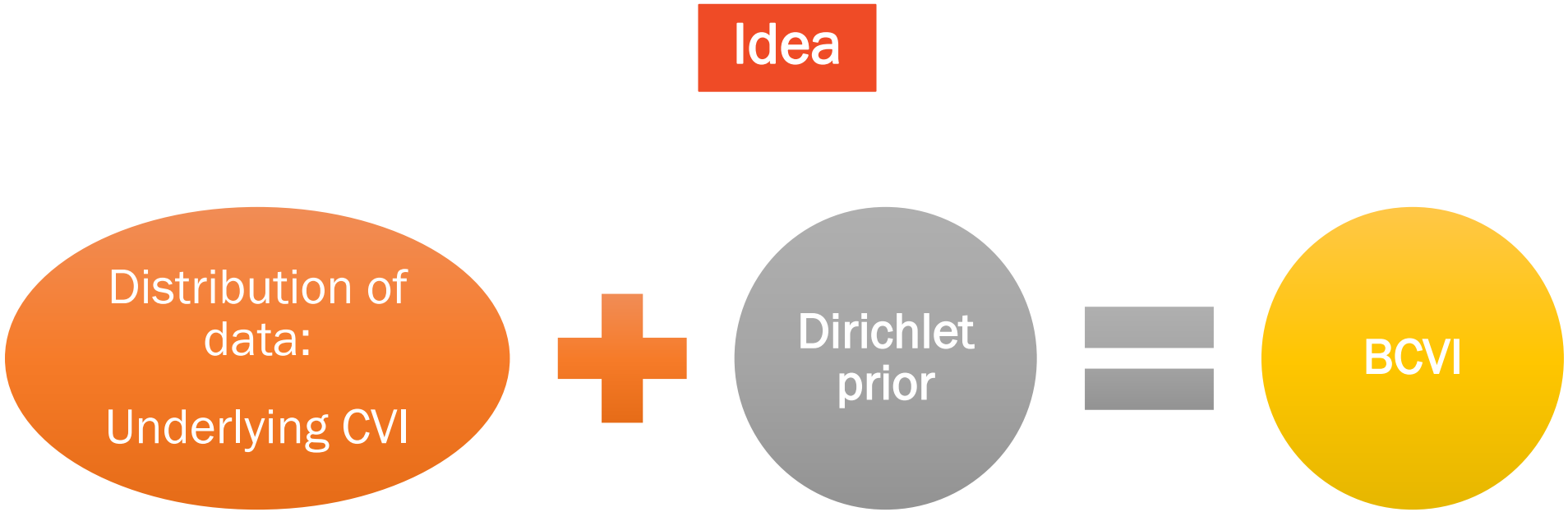
- Constantly yield several peaks which allow users to rank their options. This benefits the users in the field that the final number of clusters is flexible to choose based on their applications.





https://www.istockphoto.com/th/%E0%B8%A0%E0%B8%B2%E0%B8%9E%E0%B8%96%E0%B9%88%E0%B8%B2%E0%B8%A2/himalayan-mountain-range

Bayesian analysis and cluster validity index



Dirichlet prior and posterior

3.1.1 Dirichlet prior and posterior

Here, we assume that $\mathbf{p}$ follows a Dirichlet prior distribution with parameters $\boldsymbol{\alpha} = (\alpha_2, \ldots, \alpha_K)$ with the probability density function

$$\pi(\mathbf{p}) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{k=2}^{K} p_k^{\alpha_k - 1}. \tag{3.3}$$

Theorem 3.1 *Let $K \in \mathbb{N}$ and $\mathbf{r}(\mathbf{x}) = (r_2(\mathbf{x}), \ldots, r_K(\mathbf{x}))$, where $r_k(\mathbf{x})$ is defined as in (3.1). Assuming that $\mathbf{x}$ follows the distribution described in (3.2), the posterior distribution of $\mathbf{p}$ has the probability density function:*

$$\pi(\mathbf{p}|\mathbf{x}) = \frac{1}{B(\boldsymbol{\alpha} + n\mathbf{r}(\mathbf{x}))} \prod_{k=2}^{K} p_k^{\alpha_k + n r_k(\mathbf{x}) - 1}.$$

In particular, it follows a Dirichlet distribution with parameters $\boldsymbol{\alpha} + n\mathbf{r}(\mathbf{x})$.

Definition of Bayesian cluster validity index

Definition 3.3 *For* $k = 2, 3, \ldots, K$,

$$BCVI(k) = \mathbb{E}[p_k|\mathbf{x}] \tag{3.7}$$

where $\mathbb{E}[p_k|\mathbf{x}]$ *is computed according to either Corollary 3.1 or Corollary 3.2.*

3.3.1 Dirichlet prior

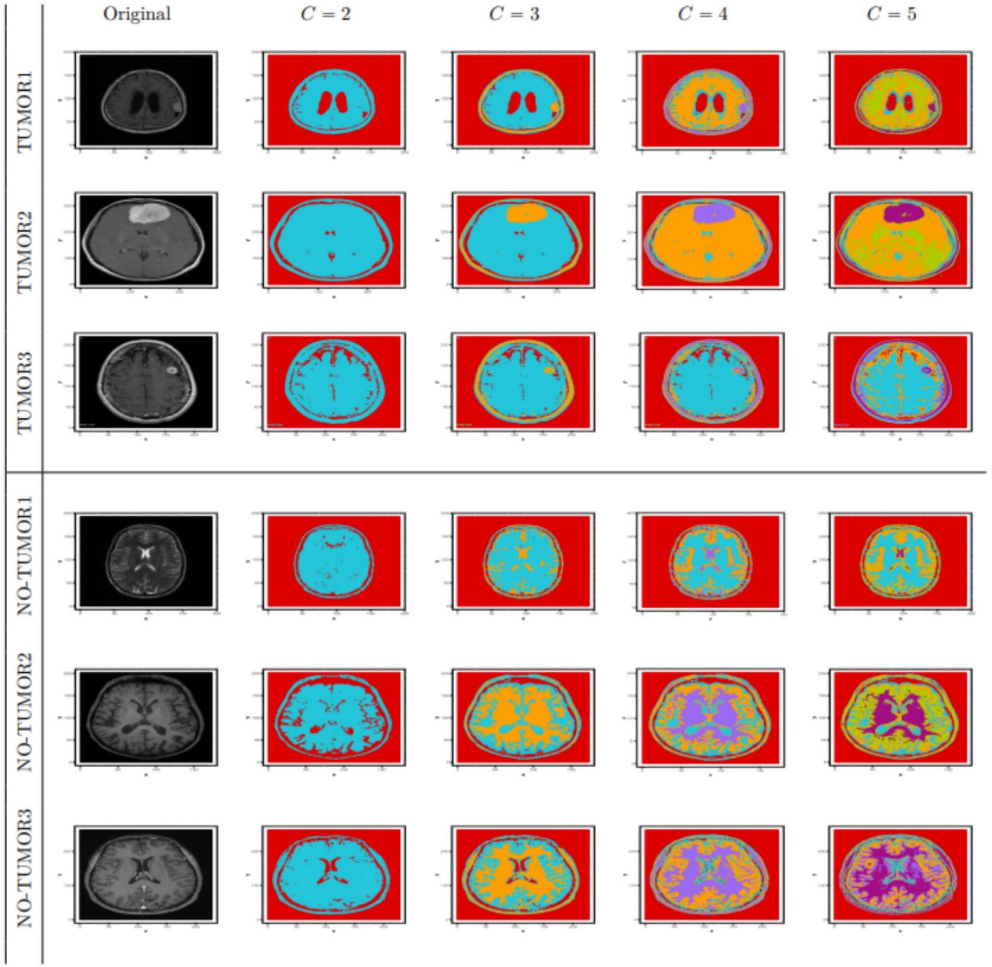
By (3.7) and Corollary 3.1, for $k = 2, 3, \ldots, K$, the BCVI is given by

$$\texttt{BCVI}(k) = \frac{\alpha_k + nr_k(\mathbf{x})}{\alpha_0 + n}. \tag{3.8}$$

The following proposition analyzes the behavior of BCVI when n is large according to the order of α_k in each situation.

Experimental results

MRI datasets



Experimental results

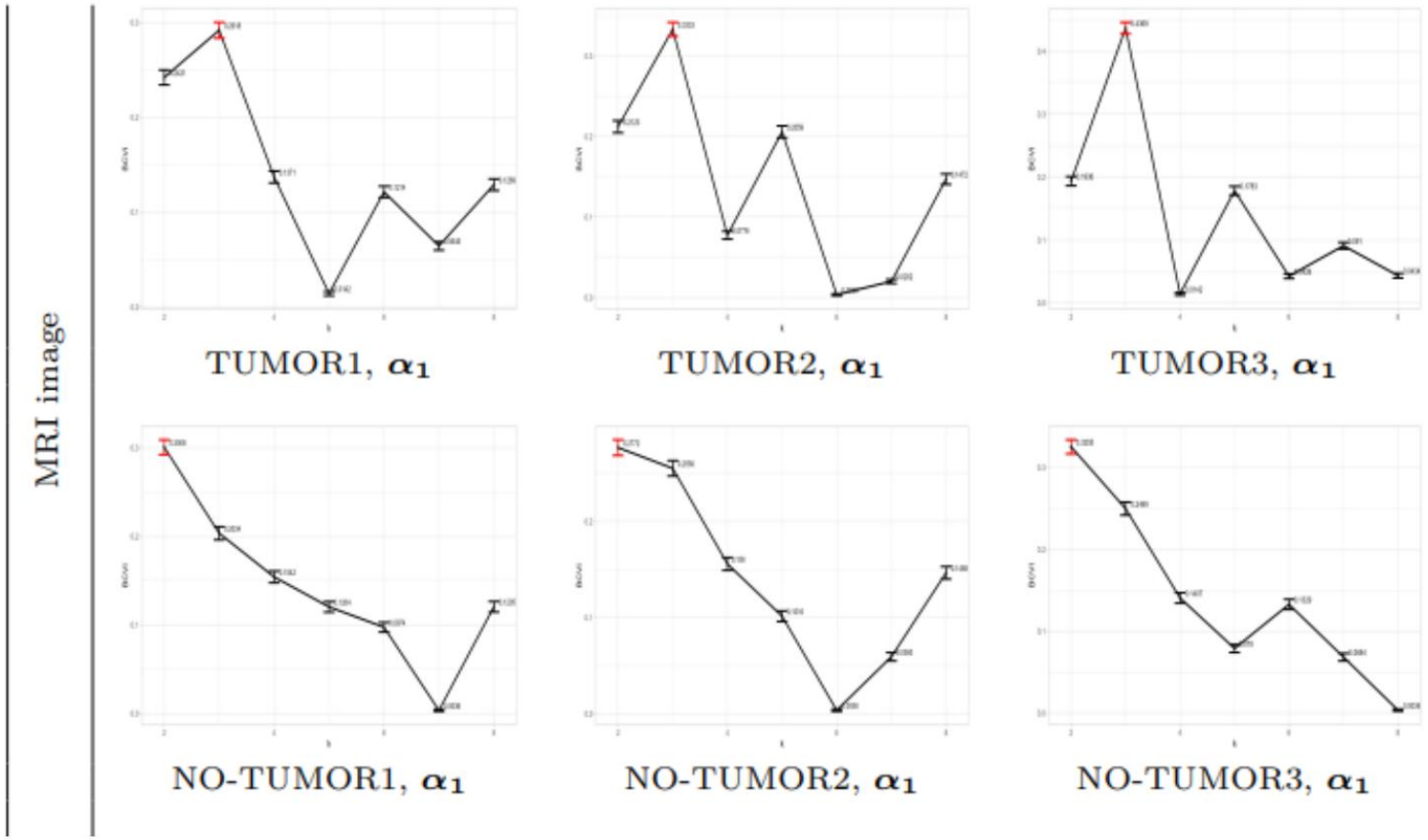
MRI datasets

Table 6: Soft BCVI on real-world and MRI datasets

Type	Data	α	K	BCVI(WP) 1	BCVI(WP) 2	BCVI(WP) 3	Porpotion of rank 1-3	WP 1	WP 2	WP 3	XB 1	XB 2	XB 3	KWON2 1	KWON2 2	KWON2 3
Realworld	SEED	1	3	**3**	2	7	81.30	7	10	5	2	**3**	4	4	**3**	2
		2		7	10	9	61.64									
		3		5	4	7	78.50									
MRI	TUMOR1	1	3	**3**	2	4	66.85	8	4	6	**3**	5	2	**3**	5	8
		2		8	6	7	71.56									
		3		4	5	8	61.65									
	TUMOR2	1	3	**3**	2	5	75.19	5	**3**	8	**3**	5	4	**3**	5	4
		2		8	5	7	61.46									
		3		5	4	3	76.21									
	TUMOR3	1	3	**3**	2	5	80.88	**3**	5	7	**3**	5	2	8	7	6
		2		7	3	8	63.49									
		3		5	3	4	77.42									
	NO-TUMOR1	1	2	**2**	3	4	65.89	4	**2**	8	3	5	7	7	5	8
		2		8	6	7	63.32									
		3		4	5	8	72.48									
	NO-TUMOR2	1	2	**2**	3	4	68.52	8	4	**2**	**2**	8	3	8	7	6
		2		8	7	6	62.78									
		3		4	5	8	73.68									
	NO-TUMOR3	1	2	**2**	3	4	71.56	**2**	6	4	**2**	4	3	6	7	8
		2		6	7	8	62.00									
		3		4	5	2	69.05									

Experimental results

MRI datasets



Highlighted Features

- **Novel and unique concept:** BCVI allows users to blend their knowledge with a dataset's pattern to identify the final number of clusters.

- **Flexibility:** BCVI allows users to flexibly set parameters according to their needs and select any clustering algorithms and underlying CVIs of their choice.

- **Correcting erroneous results:** BCVI can lead to the correct number of clusters in cases where the underlying CVI is incorrect. However, this requires users to select appropriate parameters based on their knowledge.

- **Providing alternative options:** BCVI can suggest alternative suboptimal numbers of clusters if the optimal one is not suitable for users in their context.

Drawbacks

- It relies on the quality of underlying indices.

- It is only effective when underlying indices are present, providing meaningful options for ranking local peaks for the final number of clusters.

R package: UniversalCVI

Installation

install.packages('UniversalCVI')

UniversalCVI

Use: Wvalid

```
library(UniversalCVI)

# The data is from Wiroonsri (2024).
x = R1_data[,1:2]

# ---- Kmeans ----

# Compute all the indices by Wvalid
K.NC = Wvalid(scale(x), kmax = 15, kmin=2, method = 'kmeans',
  corr='pearson', nstart=100, NCstart = TRUE)
print(K.NC)
```

Use: WP.IDX

```
library(UniversalCVI)

# The data is from Wiroonsri (2024).
x = R1_data[,1:2]

# ---- FCM algorithm ----

# Compute all the indices by WP.IDX using default gamma
FCM.WP = WP.IDX(scale(x), cmax = 15, cmin = 2, corr = 'pearson', method = 'FCM', fzm = 2,
  iter = 100, nstart = 20, NCstart = TRUE)
print(FCM.WP$WP)
```

R package: BayesCVI

Installation

install.packages('BayesCVI')

Compute BCVI for hard clustering

```
library(BayesCVI)

# The data included in this package.
data = B2_data[,1:2]

# alpha
aalpha = c(5,5,5,20,20,20,0.5,0.5,0.5)

B.WI = B_Wvalid(x = scale(data), kmax = 10, method = "kmeans", corr = "pearson", nstart = 100, sampling

# plot the BCVI

pplot = plot_BCVI(B.WI)
pplot$plot_index
pplot$plot_BCVI
pplot$error_bar_plot
```

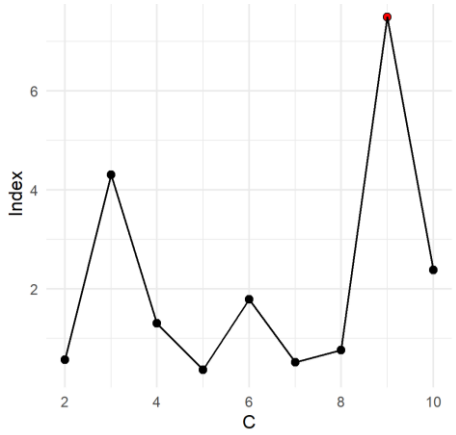
MRI brain tumor dataset

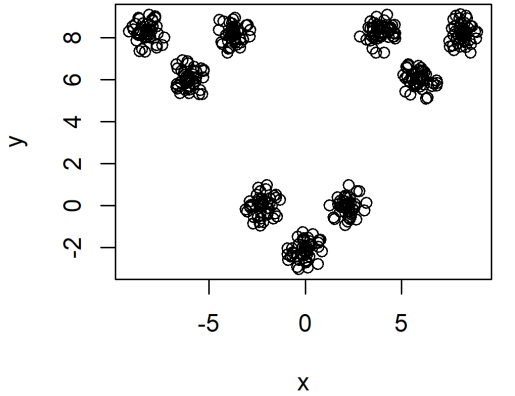
```
library(UniversalCVI)
library(BayesCVI)
library(imager)

# Download MRI data from https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-dete

x = "https://storage.googleapis.com/kagglesdsdata/datasets/165566/377107/yes/Y164.JPG?X-Goog-Algorithm=
download.file(x,'y.jpg', mode = 'wb')

IMG1 <- load.image("y.jpg")

IMG.dat = data.frame()

IMG.dat[1,"NAME"] = paste0("IMG",1)
IMG.dat[1,"DIM1"] = dim(IMG1)[1]
IMG.dat[1,"DIM2"] = dim(IMG1)[2]
IMG.dat[1,"DIM3"] = dim(IMG1)[3]

# convert to RGB

img.rgb = data.frame(
  x = rep(1:IMG.dat[1,"DIM2"], each = IMG.dat[1,"DIM1"]),
  y = rep(IMG.dat[1,"DIM1"]:1, IMG.dat[1,"DIM2"]),
  R = as.vector(get(paste0(IMG.dat[1,"NAME"]))[,,1]),
  G = as.vector(get(paste0(IMG.dat[1,"NAME"]))[,,2]),
  B = as.vector(get(paste0(IMG.dat[1,"NAME"]))[,,3]))

IMG1.RGB = img.rgb

aalpha = c(25,25,2,2,0.5,0.5,0.5)

# use sampling in function to reduce MRI image size

WP.MRI = B_WP.IDX(x = IMG1.RGB[, c("R", "G", "B")], kmax = 8, corr = "pearson", method = "FCM", fzm = 2
            nstart = 20, NCstart = TRUE, alpha = aalpha, mult.alpha = 1/2)


pp = plot_BCVI(WP.MRI)
pp$plot_index
pp$plot_BCVI
pp$error_bar_plot
```

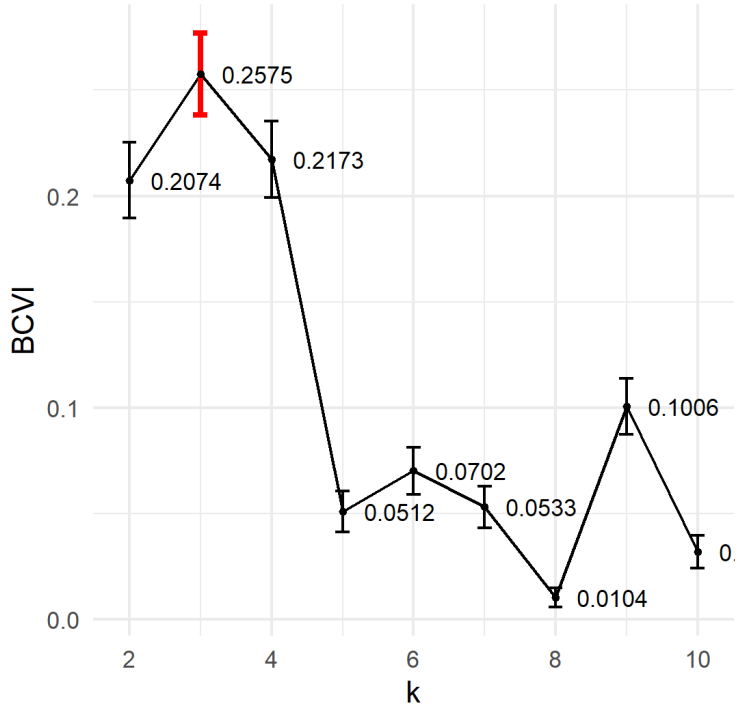
R package: BayesCVI

Example

```
library(UniversalCVI)
library(BayesCVI)

data = R1_data[,-3]
plot(data)

# Compute WP index by WP.IDX using default gamma

FCM.WP = WP.IDX(scale(data), cmax = 10, cmin = 2, corr = 'pearson', method = 'FCM', fzm = 2,
                iter = 100, nstart = 20, NCstart = TRUE)

result = FCM.WP$WP$WPI

aalpha = c(20,20,20,5,5,5,0.5,0.5,0.5)
B.WP = BayesCVIs(CVI = result, n = nrow(data), kmax = 10, opt.pt = "max", alpha = aalpha, mult.alpha = 1/2)

# plot the BCVI
pplot = plot_BCVI(B.WP)
pplot$plot_index
pplot$plot_BCVI
pplot$error_bar_plot
```







THANK YOU.

ANY QUESTIONS?