


Scientific journal - Study on the Davies-Bouldin Index

Introduction:

“ This is a notebook that tries to follow the guidelines delineated in “Ten Simple Rules for a Computational Biologist’s Laboratory Notebook”^[1].

I started it the 19/03/24, three working days since the beginning of the work, as I’m still trying to understand what it means to do research work. I’ve decided to write in English, although it’s not my mother tongue, because it only seemed appropriate since this work belongs to a bigger research that’s meant to be spread freely .

Regarding the research work on the literature in which the Index is being cited as I’m writing the journal:  The Davies-Bouldin Index in the scientific landscape ”

<https://cran.r-project.org/> to search R implementation of the index

01/10 Repeated experiments with seed setted for reproducibility:

Week 02/09-06/09, last touches:

All codes have been put up to speed. They run and do exactly what was asked of. Now that I have all the results I have analysed two of them concerning the DBI scores and here are my observations:

General considerations:

The different uses of DBI (change of intracluster distance calculation) didn’t affect the results, in fact with some basic algebra one could demonstrate that taking the centroid (and not a medoid) they are actually the same and that the differences are basically errors in calculation of the machine

In my opinion it would be useful to implement a version that uses as a intracluster measurement the variance of the cluster

Depression-HeartFailure dataset:

dbscan had a better performance here. But it’s not necessarily tied to the removal of the outliers since dbscan_small had more outliers and performed poorly, so what happened?

My opinion here is that the numerator in the DBI played the strongest role here, looking in fact to the info file one can see how the intracluster values are lower in these cases. In some way DBI is considering the structure of the dataset.

Kmeans algorithms all worked to keep the same values of distance from and between centroids, this explains why even the clusters differ much the DBI is close.

Diabetes dataset:

Hc_sing was found the most efficient by the DBI because it has put in very small clusters points that dbscan algorithms found as outliers and doing so the intercluster distance is “drugged”

Here is underlined a feature of the DBI I noticed and didn’t like, it tends to premiate imbalanced clusters.

The algorithms that had one bigger cluster had better results than others

Sepsis dataset:

The platelet count messes with the whole analysis. Only k_means attempts to divide in more than one cluster the data. DBI rewards these faulty behaviours since the intercluster measure is high (only the outliers are in other clusters) and the intracluster measure of clusters with one member is zero. It would be a good practice to construct the algorithm so that it avoids building very small clusters.

It’s notable that it’s the highest result I have seen by now and notably one wouldn’t desire

Cardiac Arrest dataset:

The data is formed by binary data and the age column. So in case of the dbscan age had the biggest role with shadings left to the other variables. Except the same behaviour with hcsing dbi treats all clusterings giving them similar results. Kmeans clustering gave very different results even so the DBI is very close for each of them

Neuroblastoma dataset:

Kmeans tends to keep the same results

Beside that no notable observation to do here

I’ve also selected some figures and tables for the paper. [Here](#)
[Here](#) the instructions to authors on peerj

August, pause:

Summer pause except I studied the course

<https://www.coursera.org/learn/sciwrite/home/module/1>

[Here](#) are my notes on it.

Weeks 15/07-26/07, researching:

These two weeks I concentrated on my last exam, anyway I managed to get some results. First I changed the scripts so that the plots have a grey background as requested by my tutor.

Second I reasoned on the mathematical sense of what happened in the matrix experiment. I'm not sure i'm there yet but I managed to understand some few things: at first I thought were involved

the [Curse of dimensionality](#) and the [Concentration of measure](#). Then I looked into the decision process kmeans selects the clusters, since it tries to minimise the variance for each cluster it makes sense it tries to put all the members in one cluster so that the variance in one cluster stays zero. My doubts remain regarding the reason behind why the problem doesn't persist till the end but has a point where it disappears. Maybe looking into the example with lower dimensions and higher numbers (or vice versa) i'll get the answer.

These days I also managed to get the output to print the results in a mannered way.

Also I started the courses on coursera to learn how to write a [paper](#)

Week 08/07-12/07:

This week I worked on reshaping the scripts to make it more comfortable to run them (i don't have to make major changes except change a number at each try) and experimented a little with the matrix experiment to understand the behaviour of k means. Also everytime I run a script there's much more information printed in output making it easier to understand the result. The behaviour of k means doesn't add up in the matrix experiment and I will investigate

Week 01/07-05/07, ending tests:

This week I finished all the tests. I committed everything on Github and now I'm elaborating my thoughts on the results of the tests. Furthermore I'm doing the cosmetical corrections on the code (mostly comments).

I will now elaborate on the steps I took. After the corrections Davide gave me I felt like I had to rewrite/adjust all the scripts. I follow line guides like atomization of the code, avoiding magical numbers and commenting thoroughly the steps I was taking avoiding repetitions. I ended up having twenty scripts in the repositories bin. Now I'll make a list of the functions with relative history:

- Loadall: I needed this script to load all my packages hastily.
- DBI and DBIndexes: first atomization i made was on the calculation of the indexes. Since the execution of `clv.Davies.Bouldin()` is a little convoluted I put it separately in the `DBIndexes()` function.

Then DBI processes its output with DBnormalize() to end up giving the four results.

- Clouds: this script has to be customised at use in the number of points in a cloud. It creates two scenarios, one with a single cloud of data and another with two clouds. Compares them then writes all the data and results on csv files. I saved the plots manually. The data gets created in two separate scripts SepClouds and SingCloud.... self explanatory
- Zero_Ones_Matrix: Has to be customised in the dimension of the space where the experiment is sustained. It creates an 2nxn matrix of n ones vectors 1xn and n zeros vectors 1xn. Then it randomises one vector at the time, alternating groups, does k-means and evaluates with DBI. All gets summarised in plot
- Labels scripts: These scripts execute the clustering algorithms with the three variations applied at each. K-means in the number of centres, hclust() in the linkage options and dbscan in a subtle change of epsilon and minimal number of points. dbscan_labels has to be customised case by case and every case is there in comment. To choose the dbscan parameters I used heuristics and the elbow method (findelbow script)
- EHRs datasets scripts: these are all the same script with minor adjustments to adapt at each dataset.
- DBI_EHRs and fixsets: this is a function that gets called by the datasets scripts and applies the DBI() function at each clustering result. With dbscan I choose to remove the outliers set with fixsets(), before evaluating the clustering.

Details [here](#).

27/06 Reorganising the work:

I'm rewriting the scripts being careful to the golden rules of programming, atomizing scripts, avoiding magic numbers and saving each result. I also restructured the repository following the instructions in this [guide](#).

And now I can commit changes to the github [repository](#) from work done locally thanks to [Github Desktop](#).

First I finished the "Clouds" script which recalls the other basic functions I saved and has to be modified for use. This script saves the data created and makes the plots

26/06 Adjourning Journal and Github:

As in the title, since I've not written anything here since 06/06.

Github's files organisation needs fixing and scripts need sanitising.

19/06 Writing code for second basic example:

As in the title. After meeting with Davide Chicco I realised I need to clean all the scripts.

12/06-13/06 Writing code for Artificial Data :

I wrote the script elaborating the 10X5 matrix example and the basic examples.

06/06 Getting accustomed to functions and starting:

Ok so, `gendata()` and `fabricate()` use in fact the `{stats}` functions that generate data using probabilistic distributions, while `syn()` extracts samples from a single dataset.

Now I'm planning writing a script where I will do the following steps:

- Generate a few datasets with `fabricate`
- Clustering with the three methods
- DBI evaluation of the previous
- Generate the NxN dataset of fixed to worsening points
- Do the Clustering and DBI evaluation
- Repete Clustering and DBI evaluation for the [datasets](#) given to me by Davide Chicco
- Graphic representation of all steps using `ggplot()`[{ggplot2}](#)

rbeta	The Beta Distribution
rcauchy	The Cauchy Distribution
rchisq	The (non-central) Chi-Squared Distribution
rexp	The Exponential Distribution
rf	The F Distribution
rgamma	The Gamma Distribution
rlnorm	The Log Normal Distribution
rlogis	The Logistic Distribution
rnorm	The Normal Distribution
rt	The Student t Distribution
runif	The Uniform Distribution
rweibull	The Weibull Distribution

05/06 Getting accustomed to clustering algorithms:

Today I tried the various functions in order to understand their requisites and their outcomes. For example [dbscan\(\)](#) seems to avoid giving a chosen number of clusters but rather gives a picture of the data landscape (lots of outliers in the dataset I generated). [kmeans\(\)](#) instead seems rudimental, my dataset is divided in half (very original). For [hclust\(\)](#) is really important the method you utilise on selecting the dendrogram (I tried single and complete). After meeting with Davide I have a better organised direction:

- First step is creating a couple of datasets one "good" and one "bad"->kmeans->DBI
- Second step is the zeros/ones dataset test->kmean->DBI->ggplot of the results
- Third step is to the EHRs datasets use 3 different configurations of kmeans(n=2,5,7), dbscan(eps/minPts) and hclust(linkage and distances)

03/06 Beginning with tests:

First I gave a quick look at the articles my tutor gave me to structure my work. Second I uploaded on Github the script normalising the DBI result.

I will use the following implementations of the clustering algorithms the test requires:

Hierarchical clustering [hclust\(\)](#) from [{stats}](#) (a foundational package)

K-Means [kmeans\(\)](#) from [{stats}](#) (a foundational package)

DBSCAN [dbscan\(\)](#) from [{dbscan}](#)

To generate the artificial datasets I'm checking the gendata() function from [{simstudy}](#), the syn() function from [{synthpop}](#) and fabricate() from [{fabricatr}](#). Also I will be using various distribution generators from [{stats}](#) (like [runif\(\)](#) or [rnorm\(\)](#))

Concerning the real datasets part I'm using those Davide Chicco [sent](#) me.

28/05-29/05 Ending table of biomedic articles:

At first I cited the techniques analysed by the CVI but then seeing the quantity of techniques appearing just once, given K-methods were the ones used the most And given the presence of the column illustrating the use made of the CVIs I preferred to not include that column.

Pathologies	Biomed problem	Cluster Scores (per appear.)
-------------	----------------	------------------------------

20%(4) Healthy 15%(3) Wide range of cancer types and subtypes 10%(2) Amputees/Upper Limb debilitated 10%(2) Acute heart failure and comorbidities 5%(1) Hypertension 5%(1) Breast cancer 5%(1) Burnout and Depression 5%(1) Dyslexia 5%(1) Colorectal carcinoma 5%(1) Chronic pain 5%(1) Diabetes and comorbidities 5%(1) Parkinson disease 5%(1) Data from a lab rat	30%(6) Improve SEMG/EMG/MES (including spike sorting) signal classification 30%(6) Improve diagnostics in respective field 15%(3) Gene analysis for cancer correlation 15%(3) Finding patterns in anamnesis to foresee outcome 10%(2) Others	45%(9) Solo 40%(8) Silhouette (SI) 15%(3) Elbow method 15%(3) Calinski-Harabasz (CHI) 5%(1) Jaccard index (JI) C-index (Hubert) Distance ratio (DRI) Bayesian Information Criterion (BIC) Dunn Index Stability Index Fisher's Linear Discriminant Index (FLDI)
Use made of the indexes	Type	
60%(12) Clustering evaluation 25%(5) Feature space performance 5%(1) Dim reduction 5%(1) Harmonisation 5%(1) Template construction	25%(5) SEMG/EMG 25%(5) Lab measurements of phenotypic expressions or symptom expressions 20%(4) Omic and Genomic data 15%(3) Questionnaires (symbolic) 10%(2) Radiomic features 5%(1) Extracellular action potential (EAP) 5%(1) Myoelectric signals (MES) 5%(1) Blood pressure features (Some articles use mixed data from these categories)	

24/05 Ending selection of [implementations](#):

As specified in the docs I followed a few steps to choose the implementation I will use:

- Download packages

- First check at source code
- Test on easy examples
- Check at documentation

Each step I found some problems with some of the implementations.

[DBindex](#) {chickn}: Its package was removed

[db_indexR](#){[SOMEnv](#)}: Isn't exacreinterpretationly DBI but a

[B_DB.IDX](#){[BayesCVI](#)}: Similar problem

[ClusterDaviesBouldinIndex](#){FCPS}: Calls on another implementation

[check_DB](#){[ulrb](#)}: The whole package is built around the biology field and its needs in terms of data reading then the function elaborates a particular format of data

[DB_weightedIdx](#){[Radviz](#)}: The package is focused in dimensionality reduction and the function elaborates its particular format of data

I won't use [DB.IDX](#){[UniversalCVI](#)} because includes an execution of the clustering algorithm which is limiting (even if it can be helpful)

This leaves me with [index.DB](#){[clusterSim](#)}, [clv.Davies.Bouldin](#){[clv](#)} and [davies_bouldin_score](#){[ClusterStability](#)}.

This is how I prepared the test:

First I prepped 6 data files containing each several vectors as written here

#Setting Artificial data

#Example 1 {A,C}*{B,D} and 2 {A,B}*{C,D}

#A<-c(1, 2)

#B<-c(2, 3)

#C<-c(3, 2)

#D<-c(2, 1)

#Example 3 {E,E}*{F,F}, 4{E,F}*{E,F} and 5

#E<-c(1, 3)

#F<-c(3, 1)

#Example 5 {E,G}*{F,H}

#G<-c(1, 1)

#H<-c(3, 3)

#Example 6 takes {B,E,G} and {D,F,H}

And i called each file Test.i (i the correlated number)

x<-Test.i

index.DB(x, cl, d=NULL, centrotypes="centroids", p=2, q=2)

y<-cls.scatt.data(x, cl, dist="euclidean")

clv.Davies.Bouldin(x, intracIs="average", intercls="centroid")#intracIs=centroid

davies_bouldin_score(x, cl)

#Ex 1, 2, 3, 4, 5

cl <-c(1,1,2,2)

#Ex 6

cl <-c(1,1,1,2,2,2)

Examples 1, 3 and 4 are pathological cases (maths sense) DB here should be +Inf, 0 and +Inf. These are thought to see how the implementations react in inappropriate conditions. Examples 2, 5 and 6 are used simply to verify if they give the correct value.

With some effort I finally executed the first test for `clv.Davies.Bouldin` (15.22) which gave the desired result (+Inf). Here's how I did it:

```
x<-as.matrix(Test.i)
x<-apply(x, 2, as.double)
x<-t(x) #all these operations on the x were necessary since if not it wouldn't read it
cl<-as.integer(cl) #obviously the same difficulty even with cl
y<-cls.scatt.data(x, cl, dist="euclidean") #passage needed for the function to work
clv.Davies.Bouldin(y, intracls="average", intercls="centroid") #eventually intracls=centroid
And here how I tested the other two:
```

```
x<-Test.i
x<-t(x)
index.DB(x, cl, d=NULL, centrotypes="centroids", p=2, q=2)
```

```
x<-as.matrix(Test.i)
x<-t(x)
davies_bouldin_score(x, cl)
```

Test results for `clv.Davies.Bouldin`:

- +Inf
- 2 (1)
- 0
- +Inf
- 2 (1)
- 2.341641 (1.369607)

Test results for `index.DB`:

- NaN
- 1
- 0
- NaN
- 1
- 1.414214

Test results for `davies_bouldin_score`:

- 0
- 1
- 0
- 0
- 1
- 1.369607

Based on these results I would choose `clv.Davies.Bouldin` and test with both intracluster distances. Since `davies_bouldin_score` gave a 0 in operations which should give me +Inf (worst result and best result shouldn't be mixed!)

And in general `index.DB` seems less accurate, both in the implementation that in the results (done by hand the result of test 6 should be 1.3696066518)

We have a winner!!

22/05 Ending first recap table and meeting with Davide:

After talking to my tutor I refined the table ended that morning. While working I found it was useless pointing out dimensionality and cardinality of the datasets (both values were high in each case). Also I considered variations as other indexes DBI was compared to.

The percentage refers to the appearances per article. (Ex. (6) means it appeared in 6 articles over 20).

Clustering algorithm	Indexes confronted	Datasets used	Intracluster measurement used
10% (2) Dim. reduc. 10% (2) C-Means 10% (2) DBSCAN 15% (3) SOM 40% (8) K-Means 45% (9) Others	10% (2) solo 15% (3) I 25% (5) CHI 30% (6) Sil 30% (6) DBI-variation 60% (12) others 65% (13) Dunn	25% (5) Mixed 35% (7) Artificial 40% (8) Real.	5% (1) Variance 20% (4) Not specified 75% (15) Centroids

20/05 Working on recap [tables](#) (index's analysis):

Today I worked on the table concerning the analysis of the papers analysing the index in order to compile a table where I recapped the most relevant aspects on the way the index was observed. I found the the following relevant aspects:

- The indexes the paper used as a comparison.
- The datasets used (whether they were based on real or artificial data).
- If real, the dimensionality of the dataset and its cardinality (useless if artificial since it would follow certain geometric rules, even the randomised ones).
- Which intracluster measurement was used (centroid distance or variance).
- The clustering algorithm (or if used for dimension reduction analysis) used.
- Whether there were some variations of the index tested.

16/05 Proceeding with [implementations](#) selection:

Ending second step of the proceeding list. Creating the theoretic test examples for the third point.

14/05 Proceeding with [implementations](#) selection:

Downloaded the packages, {chickn} was [suspended](#). And started to look at the source code.

09/05 Continuing with Davies-Bouldin's implementations in "R":

Today I've looked at the implementations of the Davies-Bouldin index I've found in order to understand which ones were peer reviewed.

[DBindex](#) {chickn}, [DavBou](#) {MGMM}, [DB_weightedIdx](#) {Radviz}, [db_indexR](#) {SOMEnv}, [DB.IDX](#) {UniversalCVI}, [B_DB.IDX](#) {BayesCVI}, [index.DB](#) {clusterSim}, [ClusterDaviesBouldinIndex](#) {FCPS}, [clv.Davies.Bouldin](#) {clv}, [check_DB](#) {ulrb}, [davies_bouldin_score](#) {ClusterStability}

07/05 - 08/05 Refinement of bibliographic research:

As Our tutor asked us to rewrite the bibliographic research done by now following a certain scheme and adding some article i've done so

03/05 Meeting with tutor after pause:

I was on holiday this week and today after the meeting i've helped my colleague Elisa with the interpretation of the gap statistic index

25/04 Studying "[R](#)":

As in the title.

23/04 Searching for implementations of D-B Index:

14:02 Searched on Google "Cran Davies-Bouldin Index". Found several implementations: [DB_weightedIdx](#), [db_indexR](#), [DB.IDX](#), [B_DB.IDX](#), [index.DB](#), [ClusterDaviesBouldinIndex](#), [clv.Davies.Bouldin](#). But most importantly I found another search engine. The "[R Package Documentation](#)": A comprehensive index of R packages and documentation from CRAN, Bioconductor, GitHub and R-Forge."

14:29 Searched on R-P-D "Davies-Bouldin", here the results:

[DBindex](#), [DavBou](#), [calcDaviesBouldin](#), [davies_bouldinC](#), [generalizedDB_fast](#), [calculate_DB_index](#), [davies_bouldin](#), [check_DB](#), [DaviesBouldinIndex](#), [iGeneralizedDB_fast](#), [davies_bouldin_score](#), [DaviesBouldinScores](#)

14:49 I looked for "D-B Index" or "DB Index" or "D-B score" or "DB score" or "D-B measure" or "DB measure": Didn't gave different results.

Since I thought I found more than enough scripts I'm starting to review them.

[DB_weightedIdx](#) {Radviz}

<https://www.rdocumentation.org/packages/SOMEnv/versions/1.1.2>

Use command `p_load` to load packages (`install.packages("name of the package") +library()`)

03/04 Continuing with the review:

10:25 Searched "Davies-Bouldin" on [Pubmed](#)

26/04 Restarting the review and helping with Gap:

Davide Chicco asked us to add more details in the analysis of the articles done by now and to add a few more articles.

23/04 and 25/04 Learning "R":

As in the title using the tutorial suggested by my tutor.

Week 15/04 to 21/04:

Except the presentation of my work on monday I had done little since I had to attend a couple of exams. Still I had the time to start to get acquainted with 'R'.

10/04/24 Ending the review (Part 2):

After preparing the presentation for the lab meeting of the 15/04/2024 I read the remaining papers. Here are my considerations on what I found:

When addressing the issue one must consider several layers, the way data are represented, the cluster algorithms one chooses to use.

The Index has advantages (is accurate, despite not discerning the cluster separation-see canberra metric, and fast) and disadvantages (it tends to favour hierarchical algorithms when compared to other clustering algorithms). There's then all the discourse tied to symbolic data, are usual clustering algorithms and scoring indexes adequate to analyse them?

A couple of considerations can be made over the performance of the Indexes, there are surely sophisticated indexes (the “I” index or Sum-of-Squares) or ways to fuse them have been explored.

I saw a couple of trials to modify Davies-Bouldin without success. (ex centre of voronoi cells) Cluster evaluation indexes can be used in various ways not necessarily tied to finding clusters: Assess a visualisation tool/a dimensionality reduction, select a feature base(?), template matching of a set of data, harmonising two data sets (here one wants the worst score).

Tha Davies-Bouldin seems a good Index but old, nevertheless some seem to not understand how indexes in general should be used....

I learned so much doing this work, both in searching the literature that in the quantity of study and discerning one must do. I also learned how clustering is a vast problem with different approaches.

09/04/24 Ending the review (Part 1):

I finished reading the articles talking about the index per se and I'm going to search for research using the index.

08/04/24 First day in the laboratory:

Getting acquainted with the lab and reading the articles analysing the index.

03/04/24 Reading articles talking about the Index:

As in the title.

02/04/24 Reading articles talking about the Index:

As in the title.

27/03/24 Quality check on articles talking about the Index:

As in the title I looked at all the articles talking about the Davies-Bouldin Index watching if I made a mistake in picking them, watching the number of citations, if the content can be helpful with the work and the evaluation of the Journal that published the article on Scimago[9] as suggested by Davide Chicco.

I deleted little less than half of the doi links I saved originally.

I then proceeded to download the articles and started to read them, prioritising the most cited. Observations on each article are reported here:

 The Davies-Bouldin Index in the scientific landscape .

26/03/24 Looking for papers and sorting them:

12:39 on google scholar I searched for strictly scientific articles with the terms: "Davies-Bouldin index" OR "Davies-Bouldin score" OR "Davies Bouldin" OR "Davies-Bouldin". This gave 420 results. I proceeded on looking for those I didn't see in the first search and repeated the process of the day before.

25/03/24 Looking for papers:

17:39 on google scholar I inserted the terms: "Davies-Bouldin index" OR "Davies-Bouldin score" OR "Davies-Bouldin"

As written in the document where all the articles are catalogued when i couldn't find immediately the doi link I searched using Google, pasting the title and the name of the researchers, other websites where the link was written. If not written in the article either I proceeded writing the first website where I found the article.

Of the 16000 results i looked at the first 1000 roughly

Next thing next I started to separate the articles based on why they cited the Index:

- Papers that analyse it
- Papers that modify it
- Papers that use it in cluster validation of their study

22/03/24 Follow up meeting with Davide Chicco:

As in the title we discussed about my work and my findings, he helped me in organising the journal better and gave me some input on how I should first look for papers: First start looking for those analysing the Index, second look at the applications. And so I should do it on monday.

21/03/24 Safety course and inquiring about distances :

Today I watched the safety course and finished the general course. At 14:30 I followed a lesson of Andrea Maurino and asked after the lesson how is addressed the problem of the distance between data: in fact I noticed how, while looking at data represented in a space with identical units of measurement there are no problems, when the dimensions in play are not with the same unit (ex. pressure and height in a patient) it gets tricky.

Maurino gave me some good inputs. First he introduced me to the concept of standardisation of the data and second he told me about techniques and different distances (from the euclidean) that are used to tackle the problem

19/03/24 Reading “Ten Simple Rules for a Computational Biologist’s Laboratory Notebook”[\[1\]](#) and starting the journal:

As in the title. I also read “A Quick Guide to Organizing Computational Biology Projects”[\[8\]](#). Between 14:00 and 16:00 I had to follow a lesson for my personal career.

18/03/24 Searching and understanding how to search:

I used google scholar[\[5\]](#) to search for articles using as keywords the ones suggested by using VOSviewer: I generated a map of the terms used most in the titles of the articles found in Web of Science[\[6\]](#) by using “Davies Bouldin Index” as a keyword.

I found an article on how to search for papers when starting a research [\[7\]](#).

I need to give more structure to my work!

15/03/24 Reading Silhouette and Starting to search papers:

As written before I read [\[3\]](#) for comparison. I downloaded VOSviewer [\[4\]](#) and learnt to use since I thought it could help me in searching for papers.

In the afternoon I attended a meeting where were discussed the thesis’s of fellow students.

14/03/24 First questions about the research:

First thing first is to try to understand what was the request. I opened [\[2\]](#) and read it, writing down definitions and searching the keywords on wikipedia just to have a taste of the boundaries of the argument. What I tried to answer to myself was:

What is the Davies-Bouldin Index? What is intended by mathematical property of the measure? What is clustering in a mathematical sense?

These are big questions and I intend to answer them in this work.

Meanwhile they made me notice a couple of things. To better understand the Davies-bouldin Index I needed to compare it to other indexes and also to keep an eye on which type of education the people who ideated the other indexes had. Davies is a doctor. This means they probably needed something more oriented to medical data. Peter Rousseeuw is a statistician and one can see that the silhouette index doesn't pay much attention on the computational cost of the method compared to the Davies-Bouldin Index which I attribute to Bouldin since he is a computer scientist.

References

- [1] "Ten Simple Rules for a Computational Biologist's Laboratory Notebook" Santiago Schnell
- [2] "A Cluster Separation Measure" DAVID L. DAVIES AND DONALD W. BOULDIN
- [3] "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis"
Peter J. ROUSSEEUW
- [4] <https://www.vosviewer.com/>
- [5] <https://scholar.google.com/>
- [6] <https://www.webofscience.com/>
- [7] "Learning to successfully search the scientific and medical literature"
Emily A. Thompson¹ & Laurissa B. Gann² & Erik N. K. Cressman
- [8] "A Quick Guide to Organizing Computational Biology Projects" William Stafford Noble
- [9] <https://www.scimagojr.com/>