Evaluating the stability of three clustering methods: FlowSOM, X-shift, and Phenograph

Steinunn Marta Friðriksdóttir & Søren Helweg Dam

December 2019

1 Introduction

Clustering algorithms like FlowSOM, X-shift, and Phenograph have been shown to accurately detect clusters of cell populations in high dimensional cytometry data that correspond to their manually gated identities.[1] The aim of this project was to evaluate how well these algorithms agree with themselves when clustering similar data with varying parameter values, i.e. evaluate their stability.

1.1 Clustering Methods

Two graph based clustering methods were chosen for this evaluation due to their claims of accuracy, X-shift[2] and Phenograph.[3] A self-organizing map (SOM) based clustering algorithm, FlowSOM,[4] was likewise chosen to be evaluated for its high speed and good performance.[1]

1.1.1 Phenograph

Phenograph is a robust clustering method developed for high dimensional single cell data. As parameters Phenograph takes in N single cell measurements and KNN to look at. It starts with finding the KNN for each cell using Euclidean distance, resulting in N sets of k neighbourhoods. Then it constructs a weighted graph from these neighbourhoods where each node represents a cell and the edges show connections to its most similar cells. The clustering is performed on the graph by detecting communities of highly interconnected nodes.[3]

1.1.2 X-shift

X-shift was developed as a fast population finding clustering algorithm for high dimensional mass cytometry data. X-shift calculates the KNN density estimate for each datapoint. It then tries to find a nearest neighbour with a higher density estimate value for each datapoint. If no such point is found the corresponding datapoint is added to a list of candidate cluster centroids. The next step is to filter out those candidates that are not true local maximum and finally puts all the datapoints directly connected to the centroid to a cluster.[2]

1.1.3 FlowSOM

The FlowSOM algorithm builds a self-organizing map (SOM) that constitutes a grid of nodes containing in the input space. The SOM is then transformed into a minimal spanning tree by connecting the nodes in a configuration that has the minimal sum of branch weights. In the final step, FlowSOM performs meta-clustering utilizing the Bioconductor tool ConsensusClusterPlus to generate the membership assignments.[4] The algorithm is quick and thus down-sampling is less crucial.[1]

2 Materials & Methods

2.1 Dataset & Preprocessing

The clusterings were performed on mass cytometry single cell data collected from 9 different cancer patients. The dataset contained 1.215.100 entries and was split into 10 parts, 1 for each patient and 1 for all patients pooled. For each part, subsamples of varying size from 10.000 cells to 100.000 cells were generated with a starting seed as defined in Section 2.2. For each starting seed, a set of 1.000 evaluation cells was generated and kept stable for 10 clustering runs. For each clustering run, the starting seed was incremented by 1 in order to randomize the remaining cells of the subsample. Cluster memberships of the evaluation cells for each of the 10 runs were extracted for analysis.

For FlowSOM, the impact of varying the clustering seed was also investigated. To do this, a variant (later referred to as Flowsom v2) of the method was developed where the entire subsample created by the initial input seed was held stable and the incrementing seed was given as input to the clustering algorithm. Because of the high speed of FlowSOM, the subsample sizes also included using the full dataset. For all results only the 1.000 evaluation cells were kept for the analysis.

2.2 Parameters

For the two variants of FlowSOM, a total of 4.000 combinations of patient, sample size, seed, and number of clusters were used as specified in Table 1.

Patients	Sample Sizes	Seeds	Number of Clusters
9 + pool	10.000, 20.000, 50.000, 100.000, All	42, 200, 404, 666, 1337	15-30

Table 1: Parameters for running the two FlowSOM variants. A sample size of "All" indicates the use of the entire dataset.

Rphenograph,[5] an implementation of Phenograph for R was run with 1.200 combinations of patient, sample size, seed and KNN were used as specified in Table 2.

Patients	Sample Sizes	Seeds	KNN
9 + pool	10.000, 20.000, 50.000, 100.000	42, 200, 404, 666, 1337	15, 20, 30, 50, 100 & 150

Table 2: Parameters for running Phenograph.

The standalone version of X-shift from the Vortex Clustering Environment[6] was used to cluster a total of 1.200 parameter sets from patient, sample size, seed and KNN as specified in Table 3.

X-shift also offers different settings for transforming the data prior to clustering. For the evaluations we kept the default settings for the first evaluation and in the second evaluation the option of scaling by standard deviation was used.

Patients	Sample Sizes	Seeds	KNN
9 + pool	10.000, 20.000, 50.000, 100.000	42, 200, 404, 666, 1337	15, 20, 30, 50, 100 & 150

Table 3: Parameters for running X-shift.

2.3 Evaluation Measures

Two measures were chosen for the evaluation of cluster stability, the Adjusted Rand Index (ARI), recommended for use when reference clusters are large and equal in size, and the Adjusted Mutual Information (AMI), which should be used when reference clusters are unbalanced, i.e. contain smaller clusters.[7] All cluster memberships for each 10 runs of the variations of the parameter sets were compared to each other using the chosen evaluation measures.

2.3.1 Adjusted Rand Index

The Rand index is a measure of the fraction of correct labelings given by a clustering method.[8] The ARI is an implementation of the Rand Index that is corrected for chance by taking the normalized difference of the Rand Index and a baseline based on a random clustering.[9,10] The ARI (R_{adj}) is defined as follows (Equation extracted from[9]):

$$R_{adj}(C, C') = \frac{\sum_{i=1}^{k} \sum_{j=1}^{l} \binom{m_{ij}}{2} - t_3}{\frac{1}{2}(t_1 + t_2) - t_3}$$

$$\text{Where } t_1 = \sum_{i=1}^{k} \binom{|C_i|}{2}, t_2 = \sum_{j=1}^{l} \binom{|C'_j|}{2}, t_3 = \frac{2t_1 t_2}{n(n-1)}$$

A value of 0 indicates independent clusterings whereas a value of 1 indicates identical clusterings.

2.3.2 Adjusted Mutual Information

Mutual information (MI) in information theory measures the agreement of two clustering assignments. A MI score close to 0.0 indicates that the two clustering assignments completely disagree and a value close to 1.0 indicates the clustering assignments agree. Since (MI) is not adjusted for chance, the value tends to increase when the number of clusters increases regardless of whether they share information or not, hence for our evaluation we utilize Adjusted Mutual Information (AMI), defined as follows:[11]

$$AMI(U,V) = \frac{MI(U,V) - E\{MI(U,V)\}}{max\{H(U,V)\} - MI(U,V)}$$

Where $MI(U,V)$ = Mutual information,

$E\{MI(U,V)\}$ = The expected value of Mutual Information &

(U,V) = The amount of uncertainty of a partition set

2.4 Computerome

To run the many specifications of the clustering methods, the supercomputer Computerome was utilized. In order to make the process run as smoothly as possible, it was decided to use job arrays, where a parameter was given as input to each run of the clustering script. This input specified a location in a parameter grid of all combination of parameters the clustering methods should run with. Thus, to exemplify, for FlowSOM a job array range of $1-4000$ was set, with 10 jobs running in parallel on a single node with 28 processors.

As output, the scripts generated an .RData file for each clustering run containing the resulting labelings of 10 repeated runs, the used lineage channels, and a variable with the used parameter settings. To make the appropriate files easy to find, their names were set in the following format: <clustering method>_<patient>_<seed>_<sample size>_<number of clusters used>.Rdata.

After the completion of all runs, the output files were analysed and the labeling results were used for computing the evaluation measures. The combined results were stored in a data frame with the following columns: Patient, Seed, sample size, number of clusters, time spent, ARI, AMI. To compute the evaluation measures ARI and AMI, the R package Aricode[12] was used. Originally, the ARI and AMI was also computed compared to the true labelings by finding the subpopulations assigned to the 1000 stable cells and compare these with those found using the clustering methods. However, the code for this has been commented out, as these measures do to a lower degree identify the internal stability of the algorithms, but instead evaluate how close the clusterings came to agreeing with that defined by manual gating. It was decided that such information was not relevant for this report.

3 Results & Discussion

All code and results can be found in the zip folders.

3.1 The Dataset

To get a feeling for the dataset, the distribution of cell types within each patient was plotted (Figure 1). For a more clean overview, it was decided to not divide the shown cell types into subpopulations. It was noted that most patients had a significant abundance in T-cells (CD8pos + CD4pos), whilst the patient *"284d2"* had a majority of natural killer cells (NK cells).

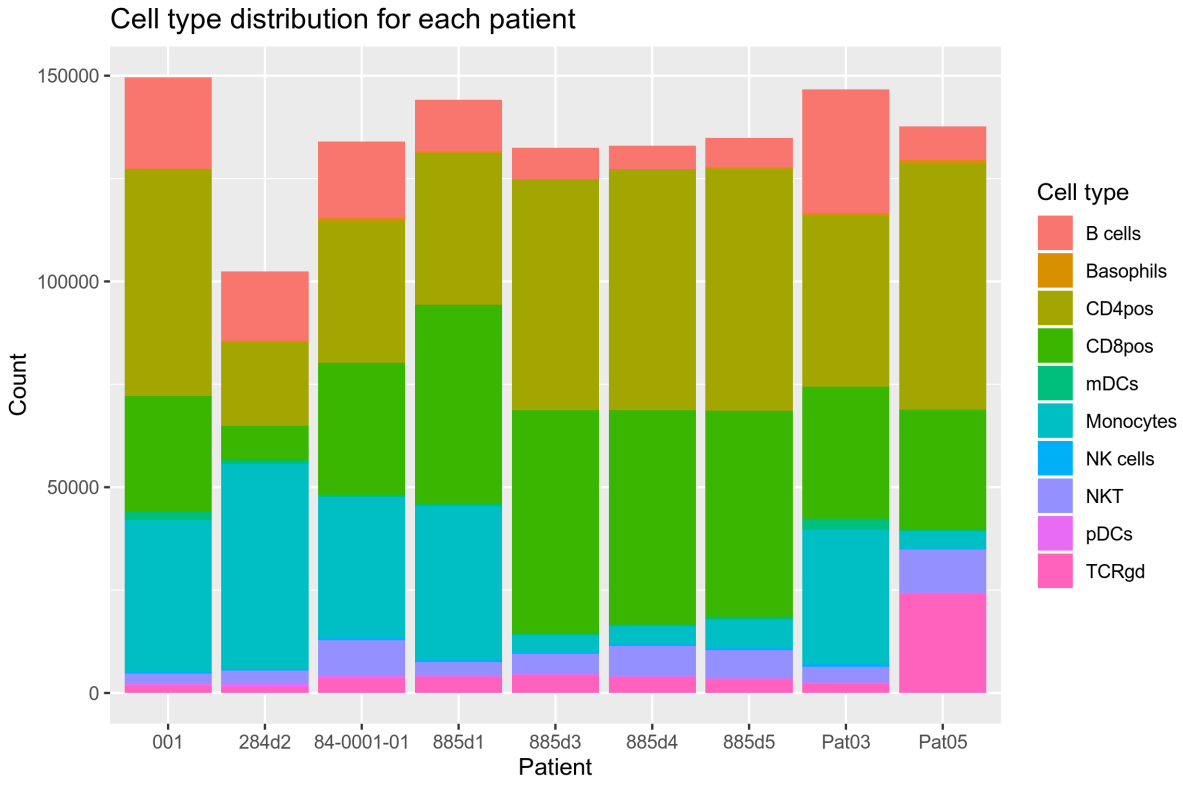


Figure 1: Distribution of cell types for each patient.

3.2 FlowSOM

Four thousand clustering runs were computed for two variants of the FlowSOM algorithm using a parameter grid of the parameters listed in Table 1. The total runtime was 898463 seconds ≈ 10.4 days, thus yielding an average runtime of 112 seconds per run, where one run constitutes 10 iterations of a clustering with incrementing seed. The longest run took around 18 minutes, where the entire dataset what used for running the clustering (patient was set to 'all' and sample size was set to 'All').

Only results from analyses on the entire dataset are shown in this report. Results at a higher detail level will only briefly be discussed and a single plot shown.

The two chosen evaluation measures (ARI and AMI) were plotted over the analysed numbers of clusters (Figure 2, panels a and b) and the sample sizes (Figure 2 panels c and d). Both measures

showed a similar trend, though AMI consistently had higher values. An interesting observation was that both FlowSOM variants performed best around the expected number of clusters (21), however, the standard error (black bars) were similar in size across all numbers of clusters, indicating the variance between the patients were equally high among all these. This variance can be confirmed, by looking at the patient-specific distributions of ARI and AMI in relation to sample size and number of clusters (results not shown). Such plots showed that there was a disagreement among the patient datasets as to what was the optimal number of clusters.

When comparing based on the sample sizes (Figure 2, panels c and d), it became apparent that the higher the sample size the better the performance of FlowSOM. This observation meant that the higher the number of cells included in the clustering runs, the more often the clustered cells occurred in the same clusters between runs.

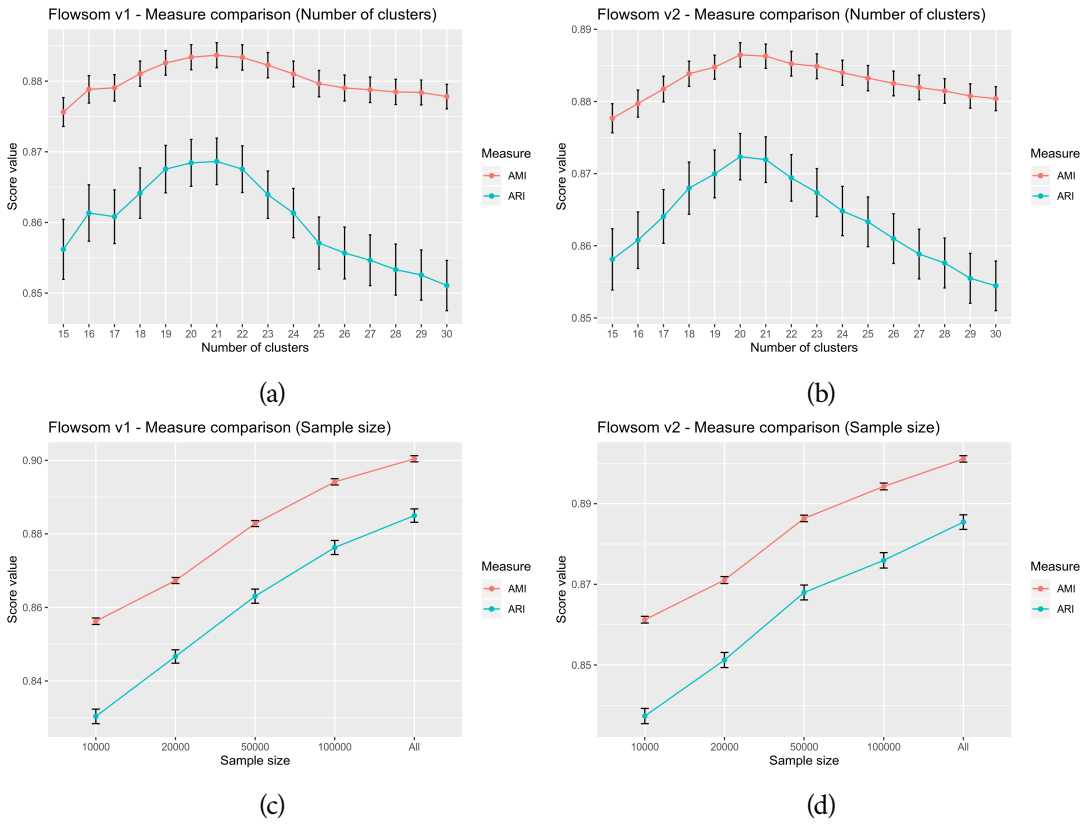


Figure 2: Comparison of measure values comparing number of clusters (panel a and b) and sample size (panel c and d). Panels a and c were made using FlowSOM v1 and panels b and d were made using FlowSOM v2. The two measures were the Adjusted Rand Index and the Adjusted Mutual Information.

To get a more detailed understanding of how these two parameters interact, the ARI and AMI values relating to the various sample sizes were plotted over the numbers of clusters (Appendix 5.1, Figure 13, all panels). A similar picture as described above was revealed, however, it became more clear that the number of clusters put into the algorithm had a greater influence with a bigger sample size. This was seen by the somewhat flat trends at the smaller sample sizes, whereas the larger sample sizes had a bump around 21 clusters. Interestingly, at lower sample sizes the performance decreased with a higher number of clusters, likely caused by the low number of cells

in each group of cell types. This could be the causation for the disagreement (i.e. high variance) described earlier.

It was investigated how the measures varied over the sample sizes among the patients (Figure 3 - only ARI for FlowSOM v2 is shown, as AMI and FlowSOM v1 gave similar results). The boxplots indicated that FlowSOM generally performed better at larger sample sizes, however, the impact varied between patients. To exemplify, the clustering performance and stability increased significantly for patient "885d1", when the sample size was increased, seen by a tighter boxplot with a higher median. Patients "284d2" and "001", on the other hand, showed a somewhat different picture. Here, the performance of the algorithm varied greatly even at high sample sizes. The reason could be the composition of cells from the patients, as patient "885d1" had a relatively even distribution of cell types, while patients "284d2" and "001" were slightly more skewed, especially "284d2", which predominantly contained NK cells (Figure 1). The fact that both variants of FlowSOM gave a similar result, revealed that both the permutation of samples given as input to the algorithm (FlowSOM v2) as well as the cells clustered with the evaluation cells (FlowSOM v1) had great influence on the performance of the method.

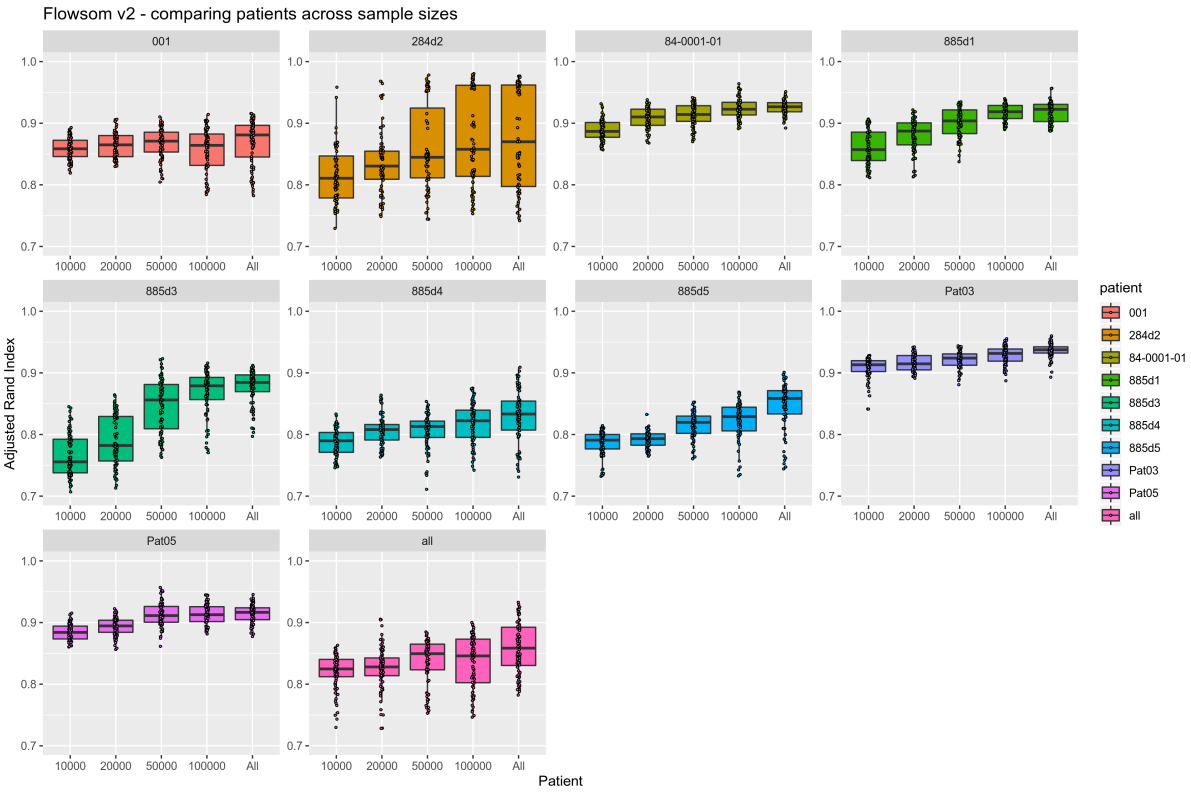
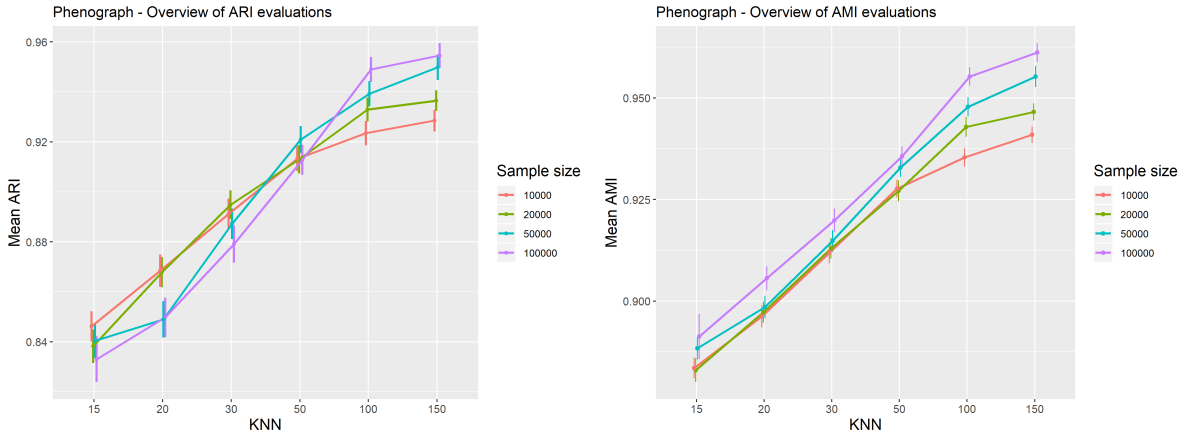


Figure 3: Boxplots of the Adjusted Rand Index for the sample sizes of the various patients in the used dataset. The patient "all" indicates the possibility of all patients to be present during clustering.

It should be noted that the results presented here are all based on mean evaluation measures of 10 iterations of a clustering run with varying seed. Thus a more in-depth understanding of the stability of the algorithms could be achieved by looking at the values individually and inspecting the variance within the 10 similar runs. In doing so, it was revealed that the ARI values varied in a span of even up to 0.3, a significant difference in labeling results (Appendix 5.1, Figure 14).

3.3 Phenograph

The results of the evaluation measures for 1.200 different parameter sets for Phenograph can be viewed in Figures 4a & 4b. The average score of ARI & AMI show similar trends where the average score for both measures increases with a larger sample size and a higher number of KNN. Both measures give relatively high scores on average but AMI scores are slightly higher.

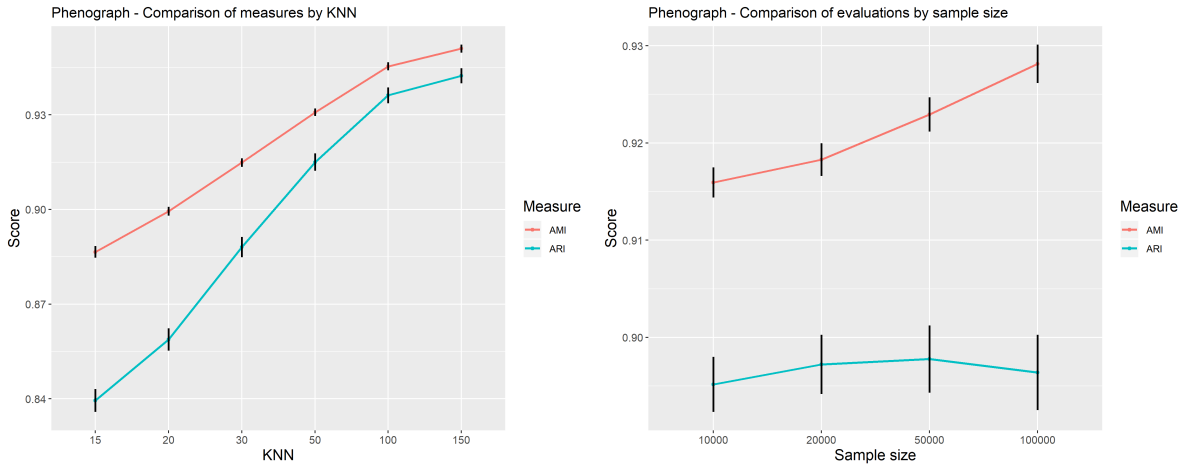


(a) (b)

Figure 4: Summary of evaluation measures for Phenograph

When evaluating the clusters based on the number of KNN (Figure 5a) there is a slight difference in scoring range where the score for AMI was a little higher on average where the number of KNN is low. The reason is most likely due to the fact that it performs better on clusters of various sizes as mentioned earlier.

The evaluations based on sample size in Figure 5b were interesting to look at as the ARI scores are quite stable over all sample sizes whereas the average AMI score increases with a larger sample size indicating the improvement of stability and self agreement with a growing sample size.

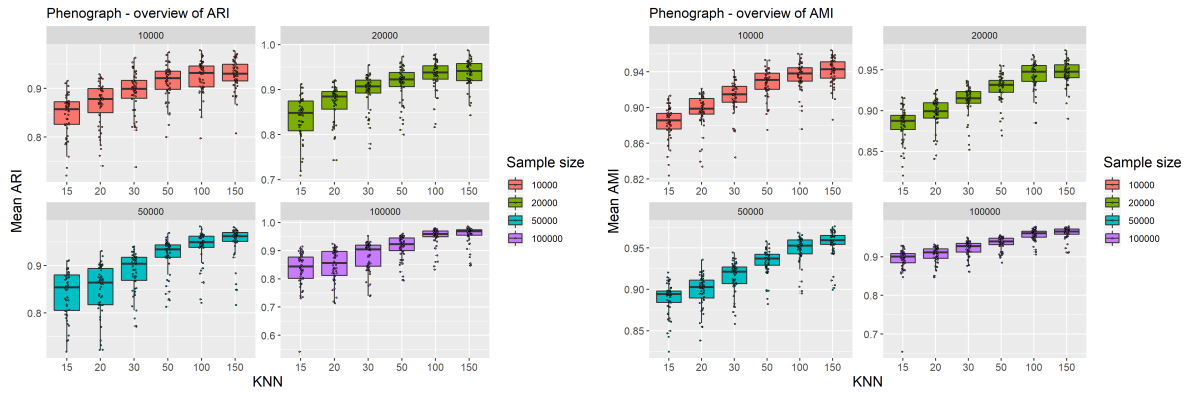


(a) (b)

Figure 5: Summary of evaluation measures for Phenograph by KNN & sample size

The summary of the evaluation measures in Figures 6a & 6b gives a very good overview of

the distribution of average evaluation scores for both ARI and AMI where both measurements seem to agree on the stability of clusterings. The larger the sample is and the higher the number of KNN the more stable the algorithm seems to be.



(a) (b)

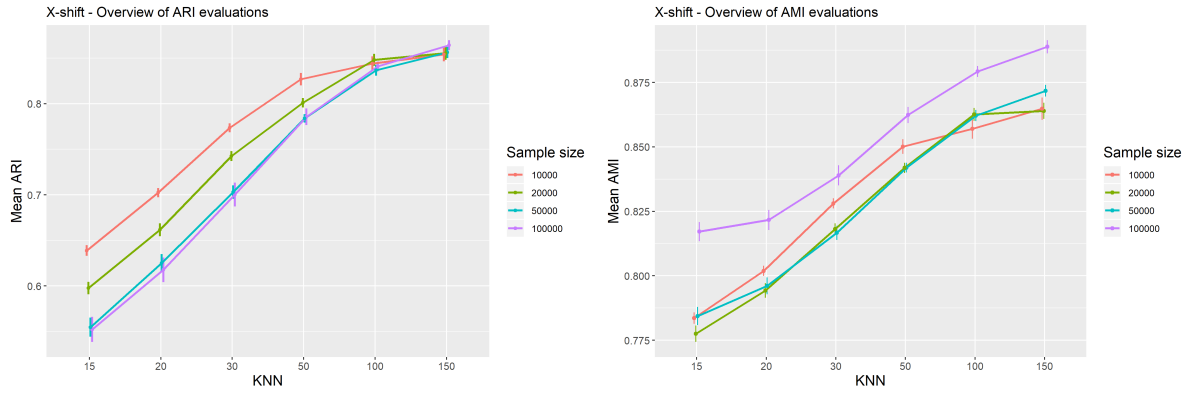
Figure 6: A boxplot summary of the evaluation measures for Phenograph

It is worth mentioning that Phenograph was quite slow with the slowest run finishing in a little over 4 hours for a sample size of 100.000 and 150 KNN.

The distribution of different number of clusters generated by Phenograph can be found in Appendix 5.2 along with more figures showing different distributions of evaluation measures.

3.4 X-shift: Default Settings

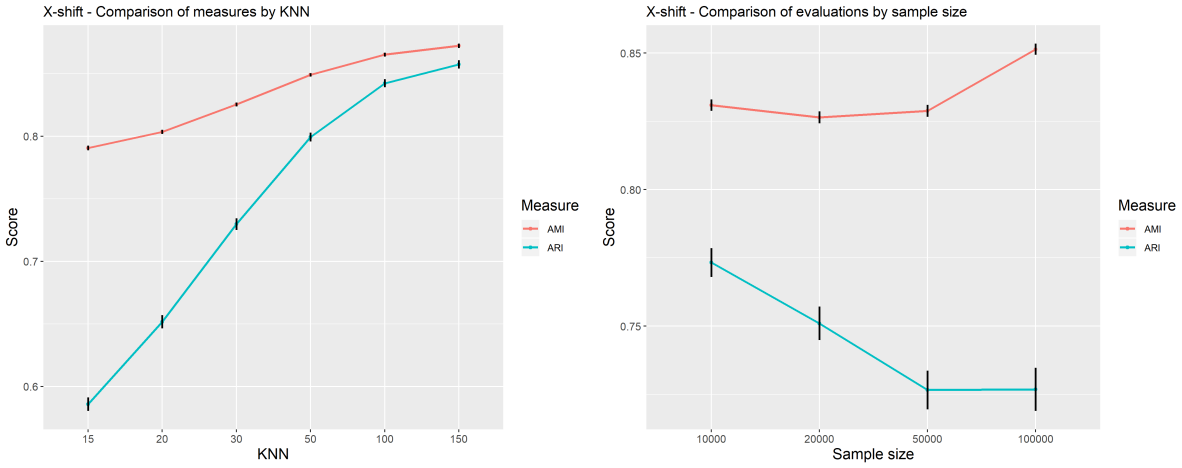
Figures 7a & 7b show the average scores of the evaluation measures performed on the 1.200 clustering results for X-shift using the default settings. The evaluation measures show a very similar trend to the evaluation of Phenograph with an increasing score on average with a growing sample size and a higher number of KNN. However the average scores for X-shift peak at similar values where Phenograph scores the lowest on average. Overall the average AMI score was higher than the average ARI.



(a) (b)

Figure 7: Summary of evaluation measures for X-shift using default settings

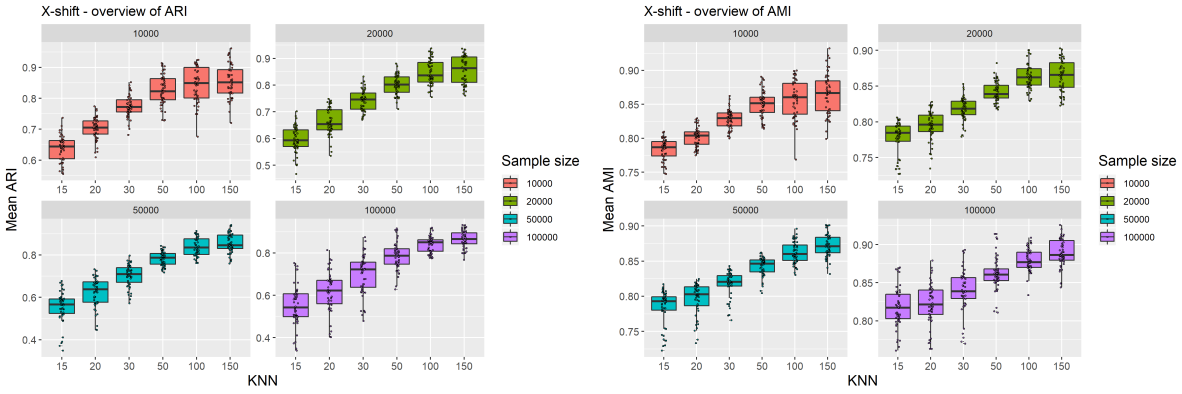
The results in Figures 8a & 8b show that the average ARI score declines when the sample size increases whereas the average AMI score improves, this could again be related to the higher number of clusters generated with a large sample size and a low number of KNN.



(a) (b)

Figure 8: Summary of evaluation measures by KNN & sample size for X-shift using default settings

The boxplots for the evaluation measures in Figures 9a & 9b do not show much of a difference in distribution between measurements but the average values of the scores differ. The ARI indicates that there is very low consistency where the value of KNN is low regardless of the sample size. AMI on the other hand shows that X-shift is quite stable for a lower number of KNN.



(a) (b)

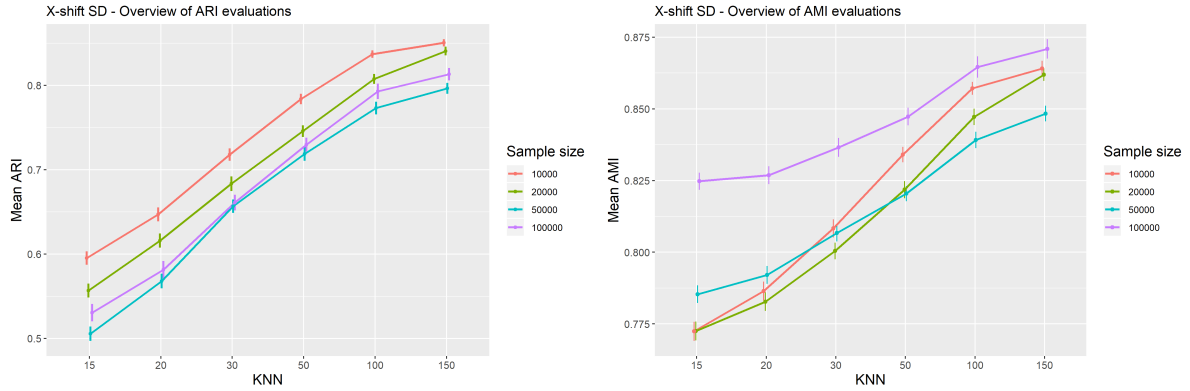
Figure 9: A boxplot summary of evaluation measures for X-shift using default settings

The distribution of different number of clusters generated by X-shift can be found in in Appendix 5.3 along with more figures showing different distributions of evaluation measures.

X-shift was reasonably faster than Phenograph where the slowest run executed in just over 30 minutes for a sample size of 100.000 and a KNN of 150.

3.5 X-shift: Re-scale by Standard deviation

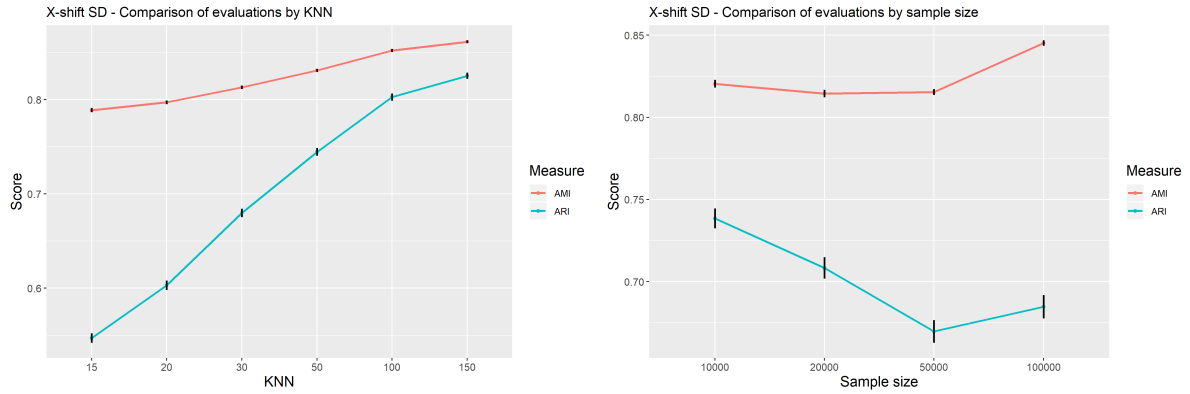
Rescaling the data prior to clustering with standard deviation did not seem to affect the evaluation as seen in Figures 10a & 10b. The distribution is very similar to what the results from X-shift using the default settings showed.



(a) (b)

Figure 10: Summary of evaluation measures for X-shift scaling by standard deviation

The results in Figure 11b again show no significant difference between the two settings of X-shift.

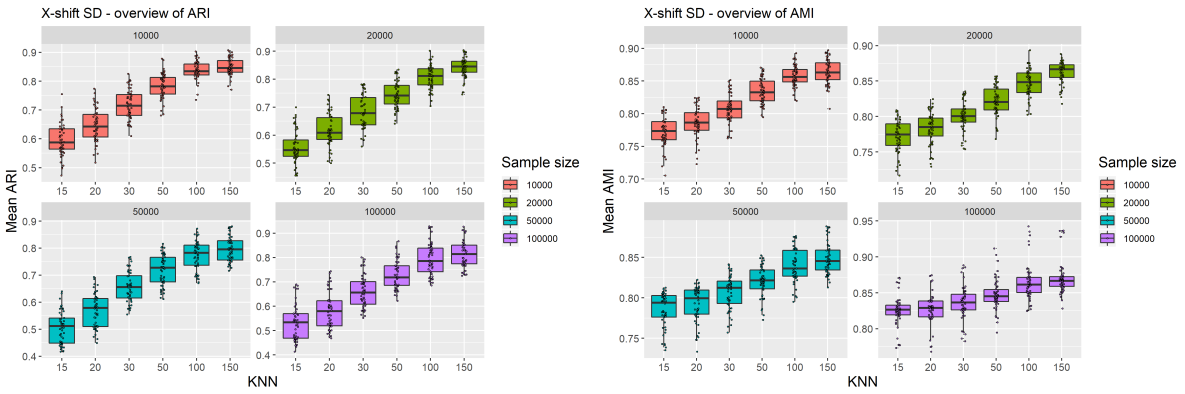


(a) (b)

Figure 11: Summary of evaluation measures by KNN & sample size for X-shift scaling by standard deviation

The boxplots for the evaluation measures in Figures 12a & 12b again show a very low ARI score on average for a low number of KNN and the algorithm appears to become more stable as the number of KNN gets higher regardless of the sample size.

Scaling the data does not seem to affect the overall results but some differences can be seen looking at the distribution of evaluation scores by patients in Figures 20, 21, 24 & 25 found in Appendix 5.4. The distribution of different number of clusters generated by X-shift with scaling can be found in in Appendix 5.4 along with more figures showing different distributions of evaluation measures.



(a) (b)

Figure 12: A boxplot summary of evaluation measures for X-shift scaling by standard deviation

4 Conclusion & Future Work

4.1 FlowSOM

Even when using all available data, FlowSOM often failed to cluster in a completely consistent manner. Although it occasionally did achieve acceptable ARI and AMI values of $\sim$ 0.9 (Patients *"Pat*03", *"Pat*05", and "84-0001-01", Figure 3), the performance of FlowSOM paled compared to Phenograph's high consistency. Therefore, when using FlowSOM, it should be considered to cluster the dataset a multitude of times with varying seeds and utilize a consensus clustering approach in order to yield a more trustworthy result. Luckily, the high speed of FlowSOM makes clustering multiple times, even at very large sample sizes, a feasible endeavor. Although it would have been interesting to confirm the claim of the importance of consensus clustering the FlowSOM results, this task was not performed in time for this report. However, the results are available for performing such an analysis and it is thus a relevant task for future work in this field.

Another interesting achievement would be to increase the performance of FlowSOM in cases where the sample is highly dominated by a few cell types, as was seen with patient *"284d2"*. One strategy that could be attempted, would be to define the largest and most stable cluster in the dataset, filter those out and recompute a new clustering result, hopefully with a more stable result. A different approach would be to tweak the weights on the used lineage channels either by utilizing machine learning to optimize this setting or by manually assigning the weights using biological knowledge. This could yield a more intelligent algorithm that uses the available knowledge about immune cells to help make more biologically valid results.

Conclusively, when working with FlowSOM it is important to be aware of the susceptibility of the method to the dataset at hand and how the seed affects the result. Further work is needed to definitively determine the best way of utilizing the fast clustering algorithm FlowSOM represents, but the highly customizable method makes a great foundation for a high-performing clustering tool.

4.2 Phenograph

Phenograph is slow but what it lacks in speed it appears to make up for in precision. It seems to be quite consistent with itself and the stability is clear when clustering a large sample with a

high number of KNN. For future work it would be interesting to run Phenograph with even more parameter sets and broadening the range of KNN and sample sizes. All in all, Phenograph appears to be very consistent with itself regardless of the sample size or number of KNN.

4.3 X-shift

It turned out to be quite troublesome running multiple instances of X-shift in parallel as it requires paths to input files to be written to the same text file which often caused errors when different instances tried writing to the file at the same time despite adding random sleep delays between writing to the input file. For future work a more optimal sleep delay should be considered in order to avoid multiple re-runs of the algorithm.

Unfortunately there was not enough time to run X-shift with more than two different settings for data transformation in this analysis. For future work on this subject it should be considered evaluating the algorithm using the different transformations the algorithm offers such as quantile rescaling or varying the scaling factor. X-shift also offers the option to search for the optimum KNN for a given dataset which was not utilized in this evaluation but it would be interesting to see if it resulted in better stability. In conclusion, X-shift does not give the impression on being very consistent with itself in comparison with Phenograph when looking at the ARI scores but AMI indicates better stability for the algorithm. The stability of X-shift seems to be dependent on a high number of KNN regardless of sample size. It did not seem to affect the result much changing the way the data is transformed prior to clustering but perhaps different scaling methods might yield more desirable results.

References

[1] Lukas M. Weber and Mark D. Robinson. "Comparison of clustering methods for high-dimen-sional single-cell flow and mass cytometry data". In: *Cytometry Part A* 89.12 (2016), pp. 1084–1096. DOI: 10.1002/cyto.a.23030.

[2] Nikolay Samusik et al. "Automated mapping of phenotype space with single-cell data". In: *Nature Methods* 13.6 (2016), pp. 493–496. DOI: 10.1038/nmeth.3863.

[3] Jacob H. Levine et al. "Data-Driven Phenotypic Dissection of AML Reveals Progenitor-like Cells that Correlate with Prognosis". In: *Cell* 162.1 (2015), pp. 184–197. DOI: 10.1016/j.cell.2015.05.047.

[4] Sofie Van Gassen et al. "FlowSOM: Using self-organizing maps for visualization and interpretation of cytometry data". eng. In: *Cytometry Part a* 87.7 (2015), pp. 636–645. ISSN: 15524930, 15524922. DOI: 10.1002/cyto.a.22625.

[5] Jinmiao Chen. *Rphenograph: R implementation of the PhenoGraph algorithm*. https://github.com/JinmiaoChenLab/Rphenograph. 2016.

[6] Garry P. Nolan Laboratory. *VorteX Clustering Environment - Java graphical tool for single-cell analysis, clustering and visualization.* https://github.com/nolanlab/vortex. 2018.

[7] Simone Romano et al. "Adjusting for Chance Clustering Comparison Measures". und. In: (2015).

[8] William M. Rand. "Objective Criteria for the Evaluation of Clustering Methods". eng. In: *Journal of the American Statistical Association* 66.336 (1971), pp. 846–850. ISSN: 1537274x, 01621459. DOI: 10.1080/01621459.1971.10482356.

[9] Silke Wagner and Dorothea Wagner. "Comparing Clusterings- An Overview". eng. In: (2010). DOI: 10.1.1.164.6189.

[10] Lawrence Hubert and Phipps Arabie. "Comparing partitions". eng. In: *Journal of Classification* 2.1 (1985), pp. 193–218. ISSN: 14321343, 01764268. DOI: 10.1007/BF01908075.

[11] Nguyen Xuan Vinh, Julien Epps, and James Bailey. "Information theoretic measures for clusterings comparison". In: *Proceedings of the 26th Annual International Conference on Machine Learning - ICML 09* (2009). DOI: 10.1145/1553374.1553511.

[12] *aricode: Efficient Computations of Standard Clustering Comparison Measures*. https://cran.r-project.org/package=aricode. Published: 2019-06-29.

5 Appendix

5.1 FlowSOM

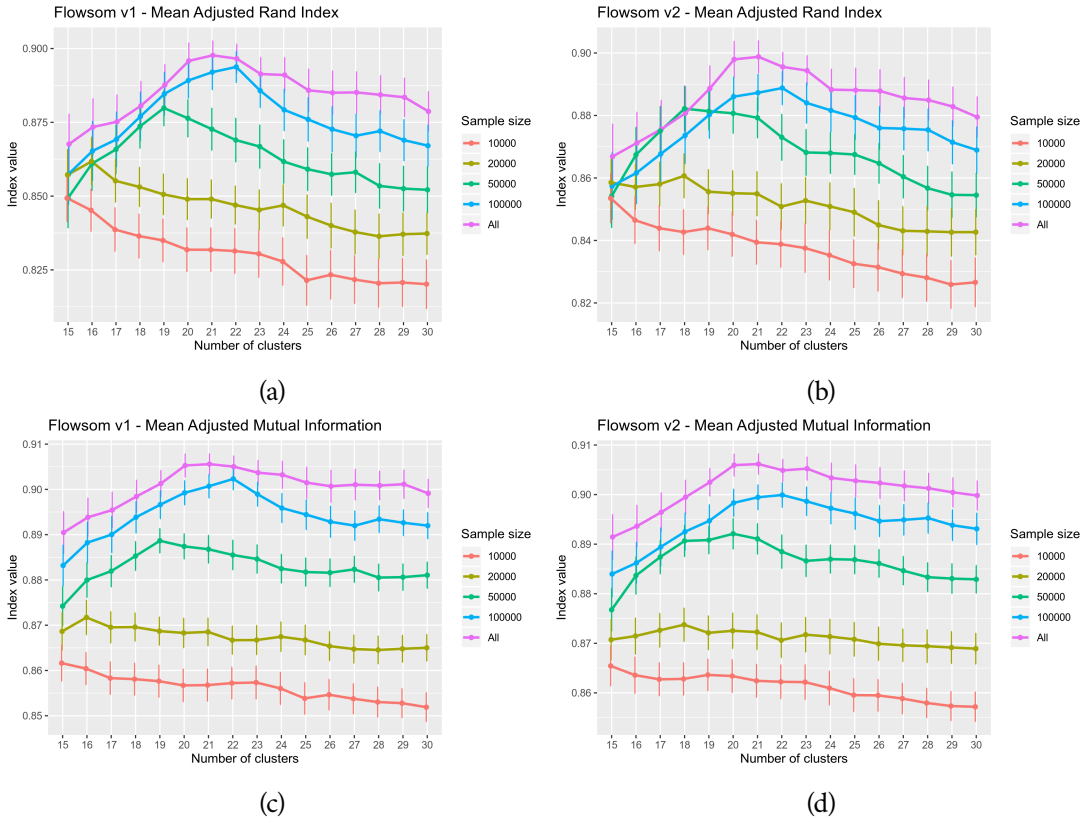


Figure 13: Change in Adjusted Rand Index (ARI) and Adjusted Mutual Information (AMI) for the two FlowSOM variants grouped by the sample size and plotted over the different numbers of clusters. Panel a is the mean ARI of FlowSOM v1, b is the mean ARI of FlowSOM v2, c is the AMI of FlowSOM v1, and d is the AMI of FlowSOM v2.

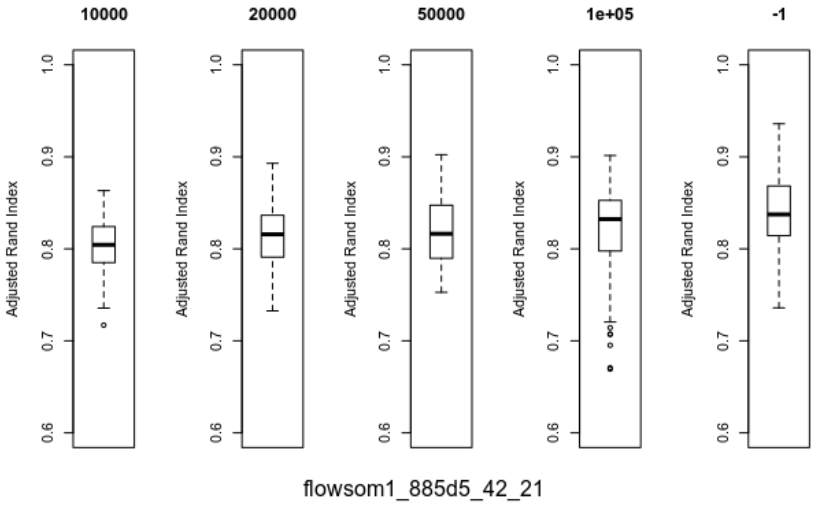


Figure 14: Boxplot of the Adjusted Rand Index values between the 10 similar runs run with FlowSOM v1, patient "885*d*5" at seed 42 with 21 clusters.

5.2 Phenograph

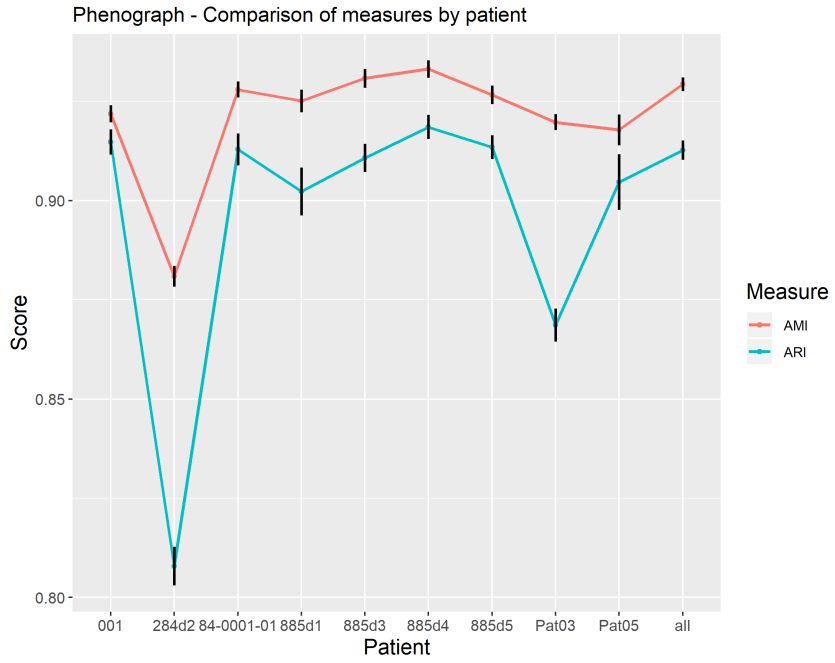


Figure 15: The average score of evaluation measures distributed by patients

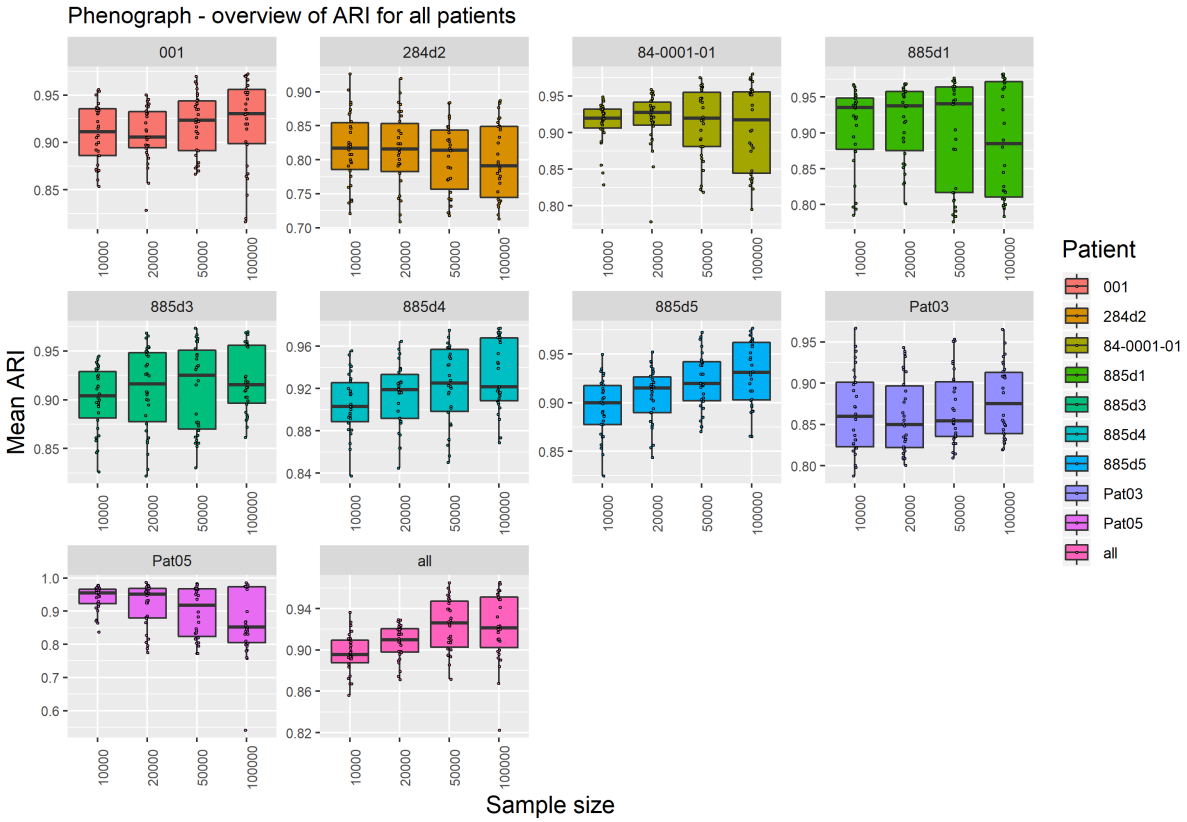


Figure 16: A boxplot summary of ARI scores for Phenograph distributed by patient & sample size

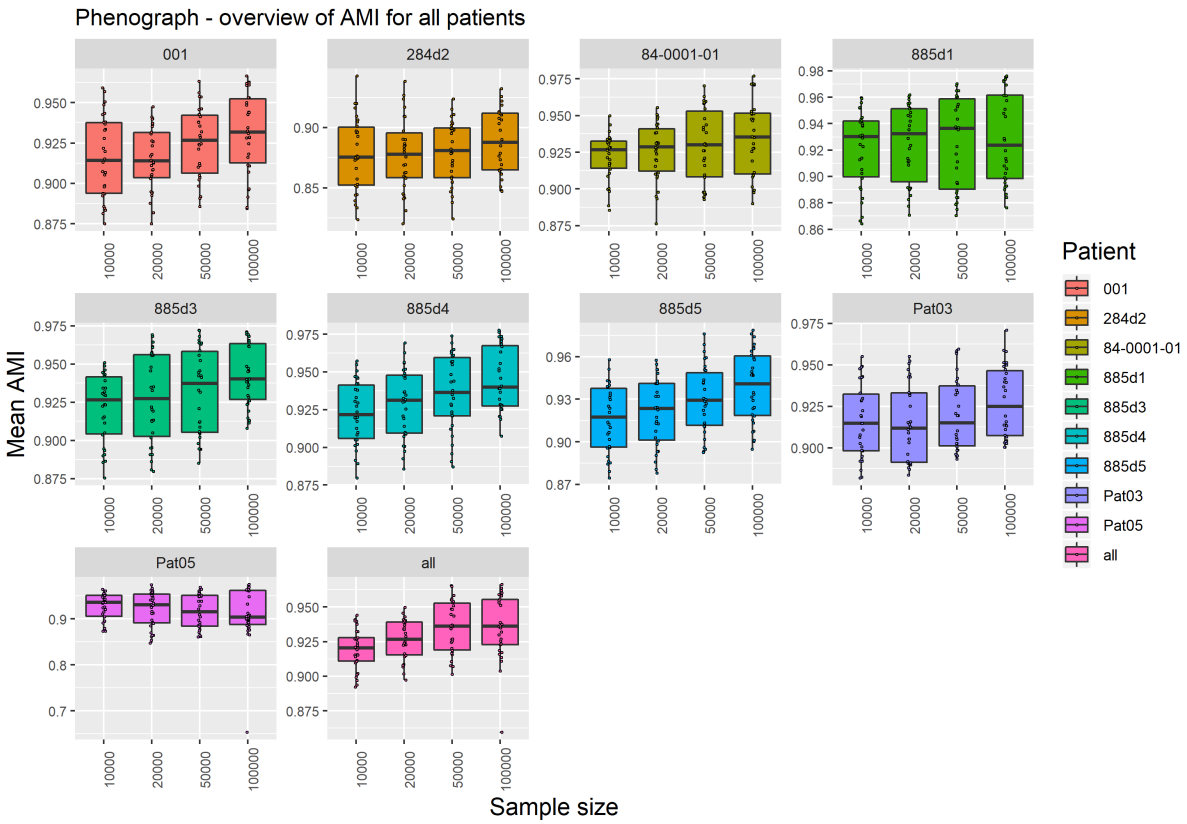


Figure 17: A boxplot summary of AMI scores for Phenograph distributed by patient & sample size

(a) (b)

Figure 18: Distribution of generated number of clusters by Phenograph

5.3 X-shift: Default settings

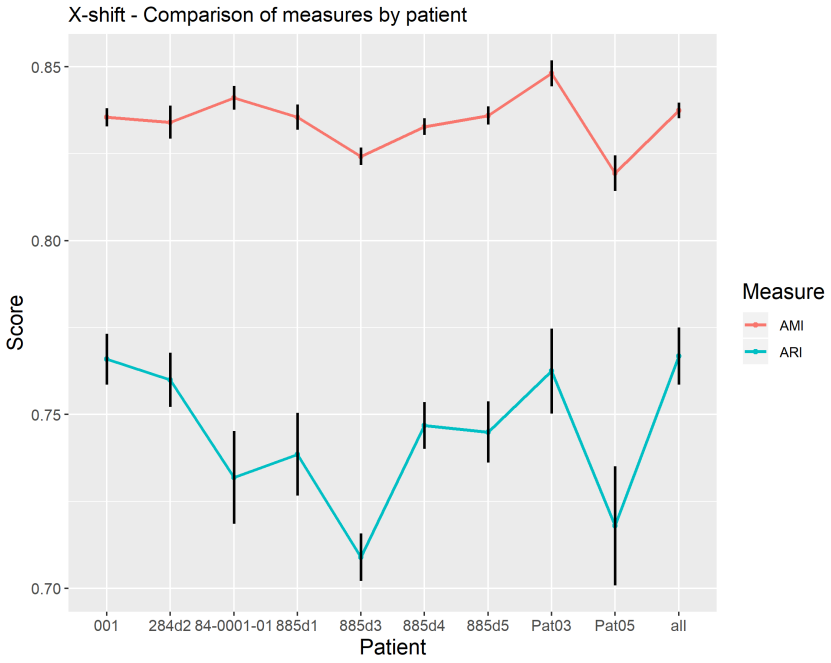


Figure 19: The average score of evaluation measures of X-shift using default settings distributed by patients

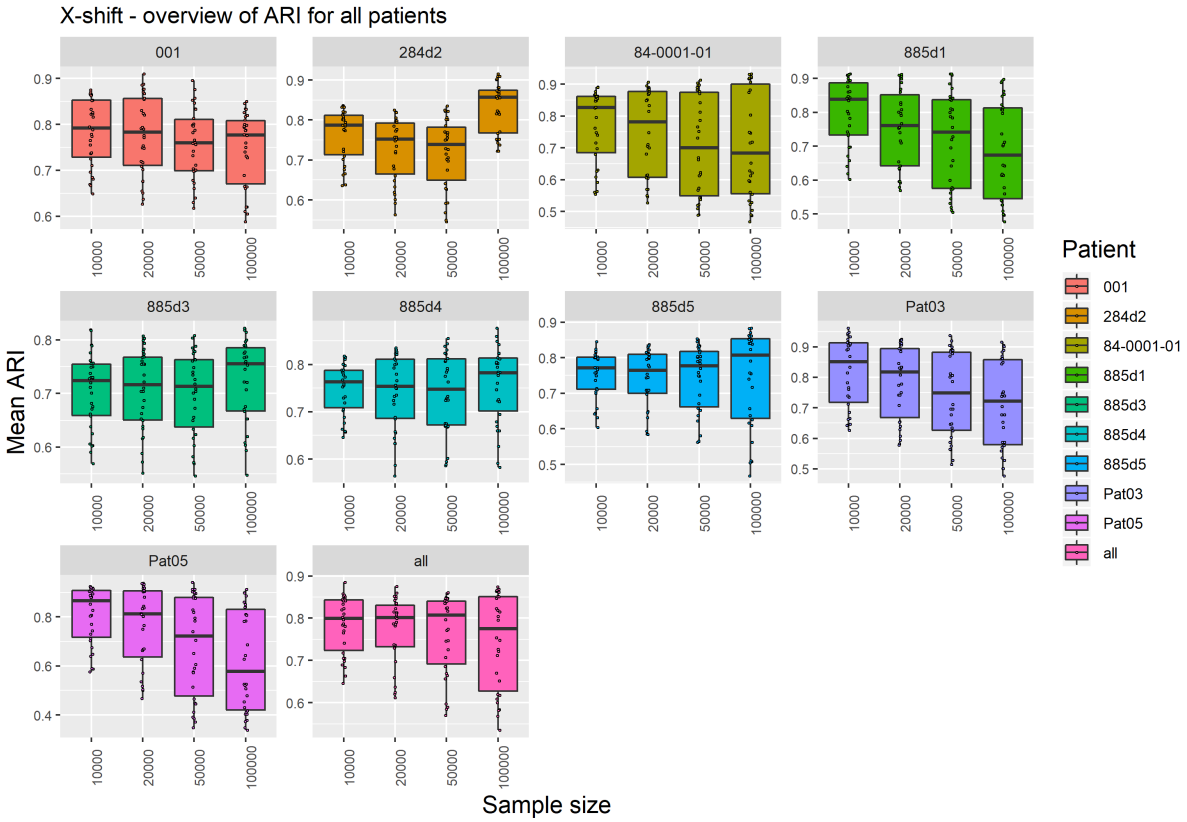


Figure 20: A boxplot summary for ARI scores of X-shift using default settings distributed by patient & sample size

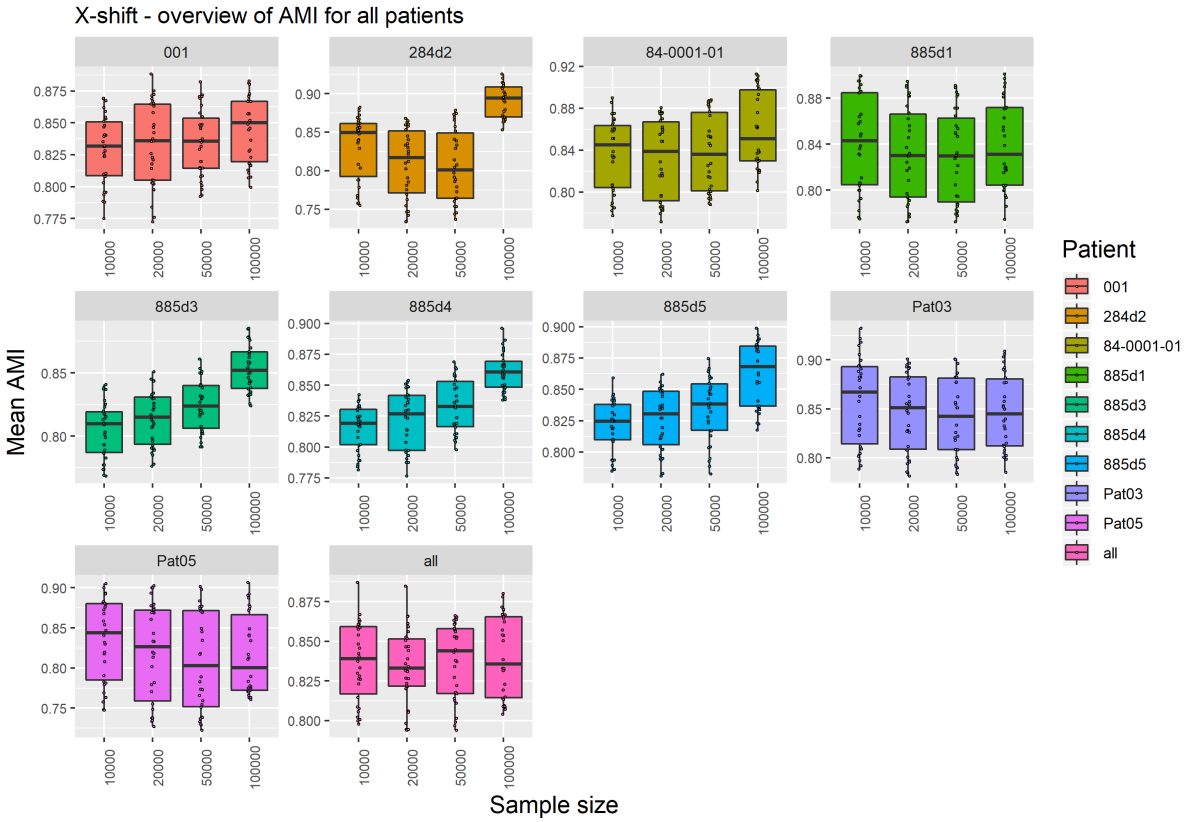
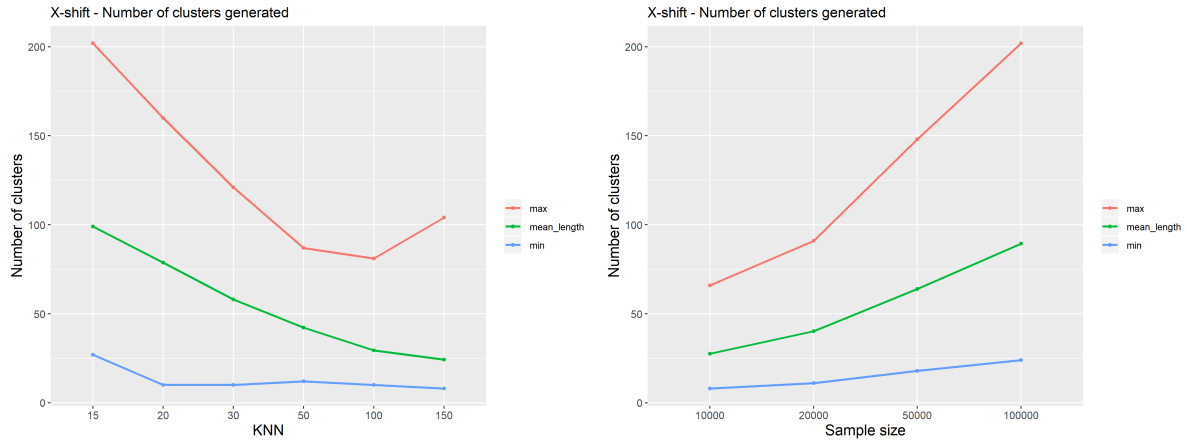


Figure 21: A boxplot summary for AMI scores of X-shift using default settings distributed by patient & sample size



(a) (b)

Figure 22: Distribution of generated number of clusters by X-shift using default settings

5.4 X-shift: Re-scale by Standard Deviation

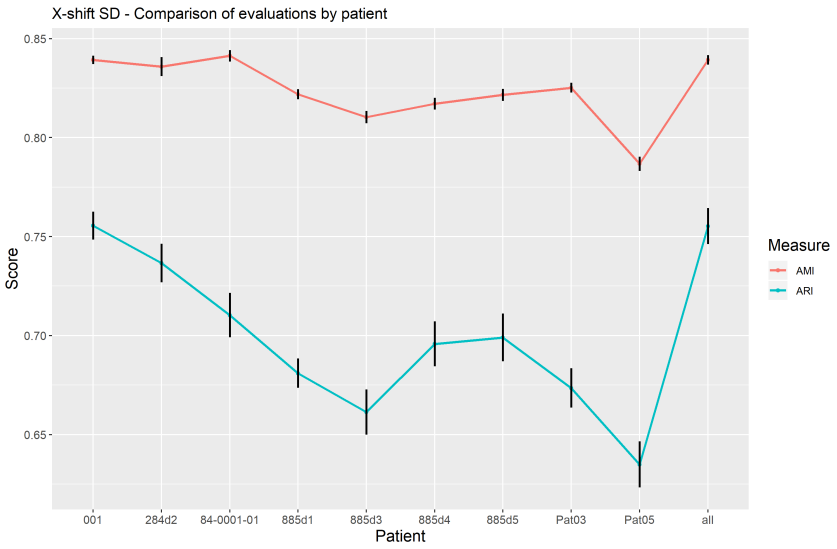


Figure 23: The average score of evaluation measures for X-shift using scaling by standard deviation distributed by patients

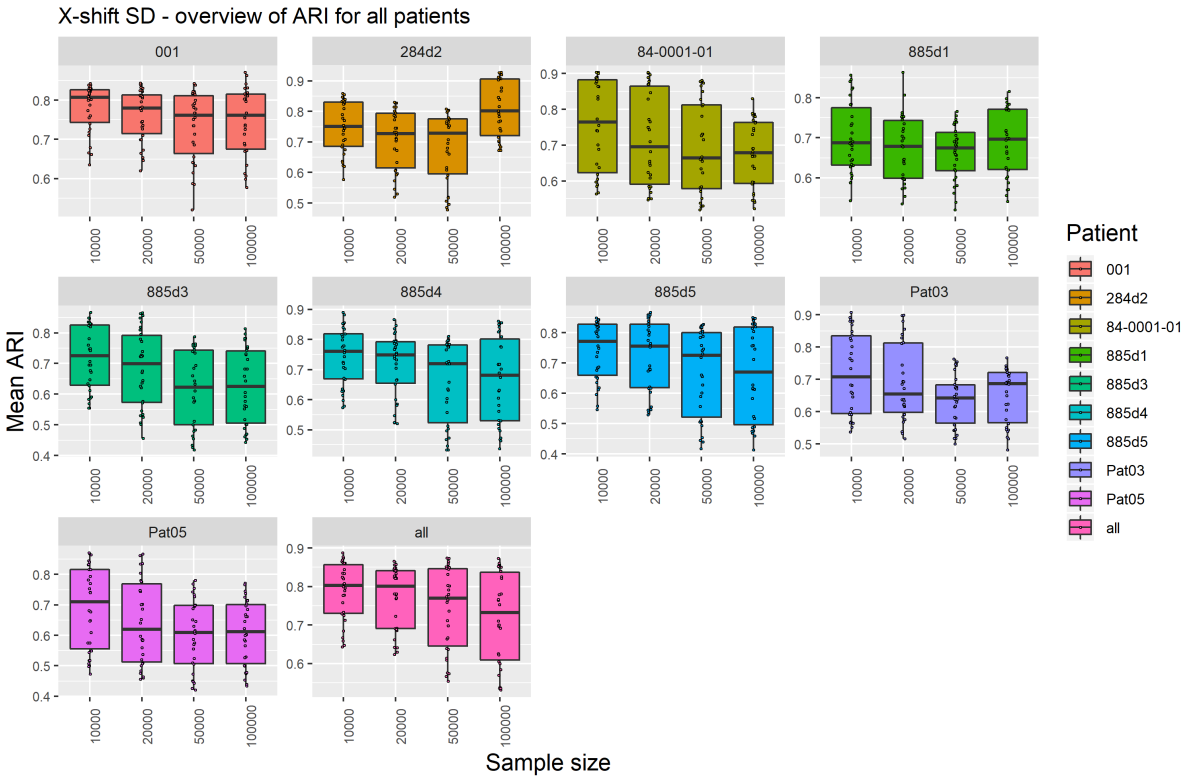


Figure 24: A boxplot summary for ARI scores of X-shift using scaling by standard deviation distributed by patient & sample size

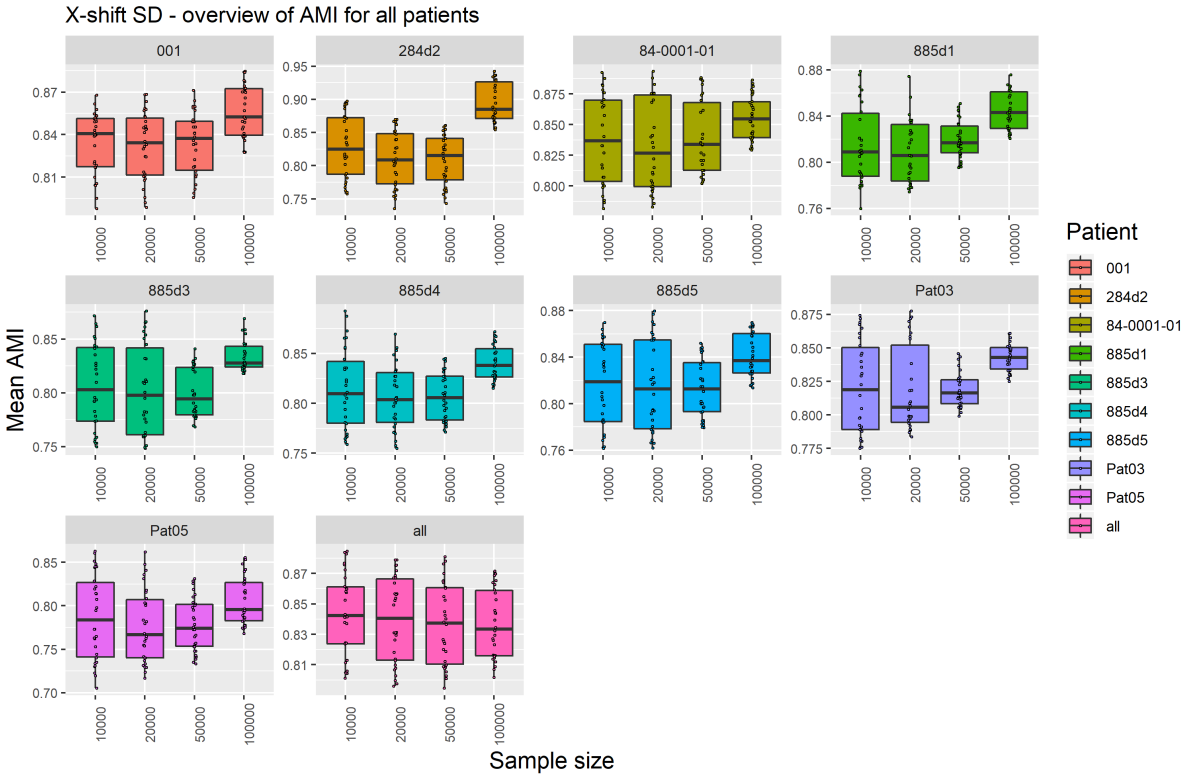
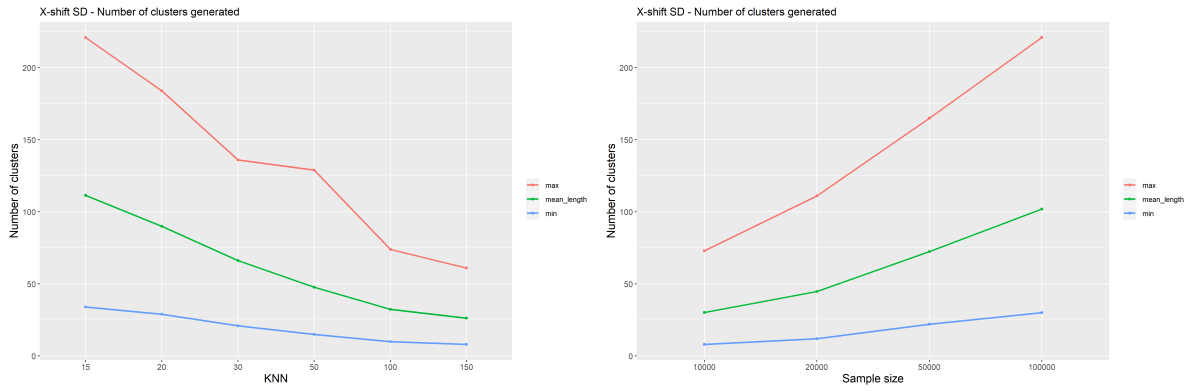


Figure 25: A boxplot summary for AMI scores of X-shift scaling by standard deviation distributed by patient & sample size



(a) (b)

Figure 26: Distribution of generated number of clusters by X-shift using scaling by standard deviation