CLUSTERING SCRATCH PROJECTS BY COMPLEXITY

Brent Meeusen | b.a.j.meeusen@student.tudelft.nl

1. Introduction

Scratch is a platform designed to learn how to program with a visual language. It is used in school curricula around the world [1]. However, it is a difficult task gaining insights in what and how children learn programming concepts. Our goal is to see if the Scratch projects can be clustered by complexity. If this is the case, we could learn more about how children learn programming.

"Could Scratch projects with a Dr. Scratch mastery score of 16 or above be clustered by different code complexity traits and project traits?"

2. Methodology

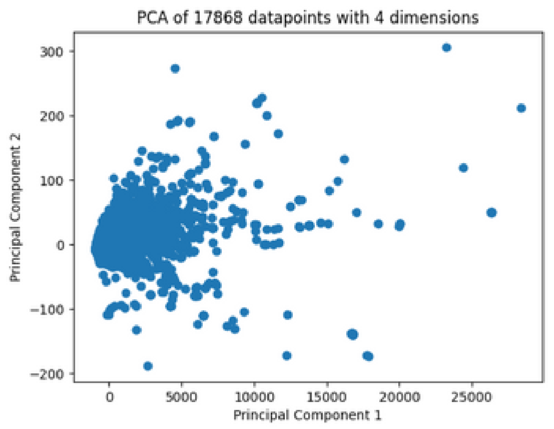
Data set: We selected a data set which contained 250,163 projects. We filtered the projects with a Dr. Scratch mastery score of 16 or above, which left us with 17,868 projects.

Clustering algorithm: We chose to implement the DBSCAN algorithm, as it is density-based, which allows it to mark outliers.

Selecting features: We selected features that could indicate the complexity of the project. We normalised some features. We also included the project names.

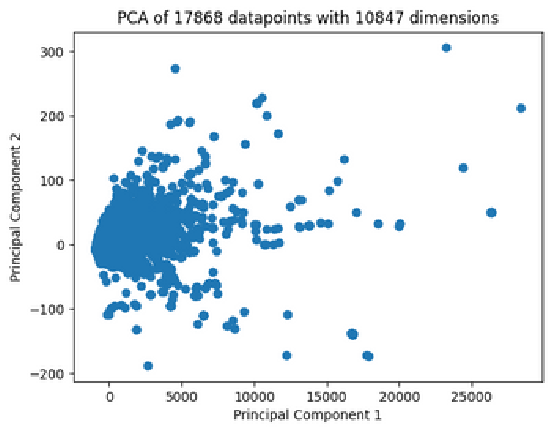
Experiments: We explored the hyper-parameters by running many different combinations and checking what works best. To measure the clusters' quality, we used the silhouette coefficient. Then, we ran five experiments using different inputs.

3. Results

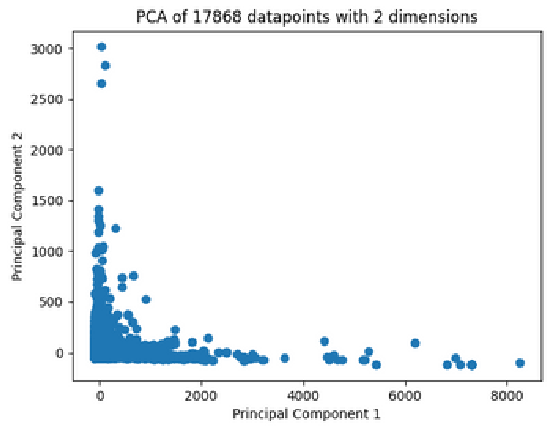


PCA: data only
Silhouette coefficient: 0.2300

These two experiments look really similar. As we can see from the graphs, there is one large cluster with some outliers and perhaps some smaller clusters around.

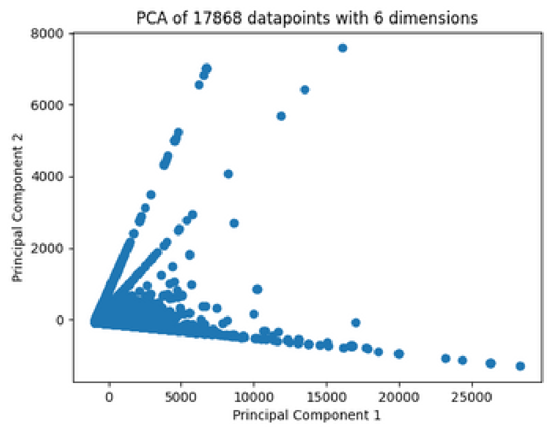


PCA: data and names
Silhouette coefficient: 0.2069



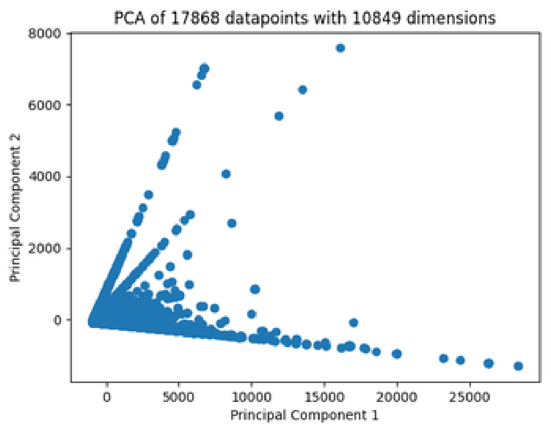
PCA: normalised only
Silhouette coefficient: 0.4078

This experiment has a large cluster in the bottom left, and has some tiny clusters going along the axes.



PCA: data and normalised data
Silhouette coefficient: -0.4731

These two experiments look really similar. We see straight line patterns in the graphs. In the bottom left, a cluster is visible.



PCA: all
Silhouette coefficient: -0.4585

On the left, we see the PCAs of the experiments with data, normalised data, and both. On the right, we see the same PCAs, but those also included the project names.
Including the project names did not have a significant impact on the outcomes. Only including normalised data did have the best cluster quality, but it also put all outliers in one cluster. Thus, we consider this result not to be trustworthy.

4. Discussion

Previous works have found clusters when only looking at the Dr. Scratch mastery score [3]. They mostly clustered projects with a low mastery score. Instead, we only included high mastery scores. In [2], Moreno-Léon et al. found that the deviation from the best fitting line increases for projects with a mastery score of 16 and above. We did not find clear clusters. Perhaps, this is because the projects are not similar enough.

5. Conclusion

We aimed to cluster projects based on their complexity to find similarities between the projects. All silhouette coefficients are below 0.50, which indicates that the cluster quality is not great. Hence, we conclude that we did not find a way to cluster Scratch projects.

References

[1] K. Falkner, S. Sentance, R. Vivian, S. Barksdale, L. Busuttil, E. Cole, C. Liebe, F. Maiorana, M. McGill, and K. Quille. An international comparison of k-12 computer science education intended and enacted curricula. *19th Koli Calling International Conference on Computing Education Research (Koli Calling '19)*, 2019.
[2] J. Moreno-Léon, G. Robles, and M. Román-González. Comparing computational thinking development assessment scores with software complexity metrics. *2016 IEEE Global Engineering Education Conference (EDUCON)*, 2016.
[3] J. Moreno-Léon, G. Robles, and M. Román-González. Towards data-driven learning paths to develop computational thinking with scratch. IEEE Transactions on Emerging Topics in Computing, 8, 2017.