Capturing the performance pattern of various students was of interest. The idea was to extract performance measures of students at multiple time steps and see if there are patterns that a cohort of students follow over the entire duration, for example, a set of students tend to start off with high scores at the start of the semester but tend to have a dip in the performance as the semester goes on.

We decided that using clustering-based approaches could serve the purpose. We found out that there were two ways of going about it:

1. **Clustering students within each timestep and tracing the cluster membership path** that every student follows over the decided time period.
2. **Clustering students considering the performance over the whole time period.**

We identified that using the first approach could give too many paths to do any sound analysis. For example, suppose we are tracing 5 clusters of students on their performance over 12 weeks (12 time steps). This can give a maximum of 5^{12} possible cluster membership paths that students can follow.

Another issue that could effect both the approaches was non-availability of assessment scores (performance measures) either because a particular student skipped the exam because of bad health, for example, or because the student is in Year 6 and we're trying to cluster him/her with Year 9 students. The latter issue can be resolved by doing a fine-grained time analysis, for a particular semester for a particular year, per say, where all the students have the same assessments. The former issue can be mitigated by simply imputing the missing value using the average assessment score for the student. There can be better ways of imputing, which will be topic of future research for us.

An interesting food for thought from the above mentioned issues is how fine-grained do we want our analysis to be - compare student performance within a semester, compare student performance across semesters, compare student performance across different year levels, or compare student performance across schools. In all but the first case, we need to compare performance scores available on a smaller time period against performance scores available on a larger time period (for example, clustering year 6 and year 10 students together). There needs to be some mapping for the performance scores in the smaller time period to the ones in the larger. We plan on using ***Dynamic Time Warping (DTW)***, an algorithm for measuring similarity between two temporal sequences. DTW is discussed in the next section.

Another design choice that has to be made is whether we trace performance of a student for every course independently or consider all the courses that a student has taken at every time step. In the latter case, a further complication is that the assessment deadlines across various courses are different. In that case, how do we define one time step? One approach can be to consider one week as a time step and aggregate all the assessment score within that week.

Proof of Concept

We tried to cluster students extracted from a very fine-grained level to avoid any complications mentioned above. The filter we used to retrieve student assessment scores are in the order:

1. campus (*GCC*) of a particular school (*JUN*)
2. year (*2018*)
3. year level (*Year 9*)
4. semester (*Semester 1*)
5. a specific class
6. a specific course (*MAT09*)

Data Cleaning

The columns that we were interested in were `result_numeric` and `result_description`. `result_description` had a mixed set of values describing the performance of every student. This included grades, for example, `A+`, `A`, `B`, etc. and also percentages, for example, `80%`, and even fractions like `27/54 (50%)`. `result_numeric` were basically just the mappings of `result_description` to percentages. However, there were many `result_description` that were not mapped to `result_numeric`. *Figure 1* lists those categories.

```
Satisfactory          1685624
Competent               87424
Absent                  81624
Not Submitted           73852
Not Satisfactory        71112
Formative Comment       62782
In Progress             42742
Not Assessed            40092
Complete                37614
Ungraded                36946
Foundation               9752
Late Submission          8870
Intermediate             3542
Not Competent            2902
Senior                   2818
Re-mark                  2746
Late Enrolment            220
Name: result_description, dtype: int64
```

***Figure 1**: `result_description` categories that were not mapped to `result_numeric`*

We used `result_numeric` for our proof of concept since we wanted to cluster performances based on numerical scores. Further, we dropped data points containing NA values and also duplicates.

Next, using the cleaned subset of data, we created a data-frame tuned to our task of clustering, where each row corresponds to a student and each column represents an assessment score. The columns were sorted by submission due dates of the assessments to do a time-series analysis. *Figure 2* shows the initial data-frame and our task-specific data-frame extracted from the former.

	stu_uuid	course_uuid	course_code	ttclass_uuid	class_uuid	class_code	class_name	class_description	assessment_uuid	assessment_result_uuid	result_numeric	result_description	submissions_due
0	1a9f40bd-1de2-a9b1-32eb-99f64c524e63	1a9fa0cc-63b2-ea9c-c33f-bdc58744bfea	MAT09	1aa845d1-bf87-7551-1c62-aa47b42710fa	1a9ca938-2286-e044-738e-3f39b320171d	MAT09:D	MAT09:D	Year 9 Mathematics:D	1aa9b816-5f0e-0a1d-c293-c03391a202af	1ac96294-e49c-693e-61b4-6a2f7ef8d80c	0.64	C	2018-11-12T12:59:59+00:00
1	1a9f87fa-8f6b-f62a-1f1f-b941b7631b87	1a9fa0cc-63b2-ea9c-c33f-bdc58744bfea	MAT09	1aa845d1-bf87-7551-1c62-aa47b42710fa	1a9ca938-2286-e044-738e-3f39b320171d	MAT09:D	MAT09:D	Year 9 Mathematics:D	1aa9b816-5f0e-0a1d-c293-c03391a202af	1ac96543-ef6b-f786-7ce7-350326a5b832	0.56	D+	2018-11-12T12:59:59+00:00
2	1a9f178e-1258-1732-06bd-02d7e5059857	1a9fa0cc-63b2-ea9c-c33f-bdc58744bfea	MAT09	1aa845d1-bf87-7551-1c62-aa47b42710fa	1a9ca938-2286-e044-738e-3f39b320171d	MAT09:D	MAT09:D	Year 9 Mathematics:D	1aa9b816-5f0e-0a1d-c293-c03391a202af	1ac9677b-bc06-e944-30da-eeabb8459900	0.64	C	2018-11-12T12:59:59+00:00
3	1a9fa7f8-ac7a-8b52-1c47-7ab4faf9353b	1a9fa0cc-63b2-ea9c-c33f-bdc58744bfea	MAT09	1aa845d1-bf87-7551-1c62-aa47b42710fa	1a9ca938-2286-e044-738e-3f39b320171d	MAT09:D	MAT09:D	Year 9 Mathematics:D	1aa9b816-5f0e-0a1d-c293-c03391a202af	1ac9358f-32a3-78db-b5c2-b27f5c9aa1b0	0.72	B	2018-11-12T12:59:59+00:00
4	1a9fc4b7-bcb1-8724-ce07-cc3a487ff637	1a9fa0cc-63b2-ea9c-c33f-bdc58744bfea	MAT09	1aa845d1-bf87-7551-1c62-aa47b42710fa	1a9ca938-2286-e044-738e-3f39b320171d	MAT09:D	MAT09:D	Year 9 Mathematics:D	1aa9b816-5f0e-0a1d-c293-c03391a202af	1ac9b2ad-2e99-fec1-91fc-a81e47a303a1	0.52	D	2018-11-12T12:59:59+00:00
...	...	...	...	...	...	...	...	...	...	...	...	...	...

Initial data-frame

	stu_uuid	2018-08-03T12:59:59+00:00	2018-09-21T12:59:59+00:00	2018-11-12T12:59:59+00:00	2018-11-23T12:59:59+00:00	2018-11-24T12:59:59+00:00
0	1a9f40bd-1de2-a9b1-32eb-99f64c524e63	0.77	0.600000	0.64	0.97	NaN
1	1a9f87fa-8f6b-f62a-1f1f-b941b7631b87	NaN	0.600000	0.56	0.28	NaN
2	1a9f178e-1258-1732-06bd-02d7e5059857	NaN	0.714286	0.64	0.48	NaN
3	1a9fa7f8-ac7a-8b52-1c47-7ab4faf9353b	NaN	0.714286	0.72	0.68	NaN
4	1a9fc4b7-bcb1-8724-ce07-cc3a487ff637	NaN	0.628571	0.52	0.28	NaN
5	1a9fb453-40fc-5369-9a30-fc6fb2e68b9f	0.48	0.828571	0.64	0.97	NaN
6	1a9f6038-dc95-3c44-91df-90ed91ab6050	0.28	0.828571	0.72	0.68	NaN
7	1a9fda63-96a2-9216-c816-13c890bf35cd	0.97	0.800000	0.72	0.97	NaN
8	1a9f08a2-476c-a8ea-1e67-f7aa6f2d1f73	NaN	0.514286	0.28	NaN	NaN
9	1a9fde9d-f4c6-07fd-adf2-865506243cfb	0.97	0.685714	0.64	0.68	NaN
10	1a9f9118-76c7-b7c8-af03-630692a9cf81	0.97	0.771429	0.72	0.68	NaN

Extracted data-frame for clustering. Columns are assessment due dates sorted in order of date-time.

Figure 2*: The initial data-frame and our task-specific data-frame extracted from the former.*

The extracted data-frame had some `NaN` values - meaning, that assessment scores for that particular student wasn't available. It can be because the student skipped the assessment for some reason or the score wasn't recorded. We imputed these missing values with the mean value of assessment scores for that student. *Figure 3* shows the data-frame with the imputed values.

	2018-08-03T12:59:59+00:00	2018-09-21T12:59:59+00:00	2018-11-12T12:59:59+00:00	2018-11-23T12:59:59+00:00	2018-11-24T12:59:59+00:00
stu_uuid					
1a9f40bd-1de2-a9b1-32eb-99f64c524e63	0.770000	0.600000	0.64	0.970000	0.745000
1a9f87fa-8f6b-f62a-1f1f-b941b7631b87	0.480000	0.600000	0.56	0.280000	0.480000
1a9f178e-1258-1732-06bd-02d7e5059857	0.611429	0.714286	0.64	0.480000	0.611429
1a9fa7f8-ac7a-8b52-1c47-7ab4faf9353b	0.704762	0.714286	0.72	0.680000	0.704762
1a9fc4b7-bcb1-8724-ce07-cc3a487ff637	0.476190	0.628571	0.52	0.280000	0.476190
1a9fb453-40fc-5369-9a30-fc6fb2e68b9f	0.480000	0.828571	0.64	0.970000	0.729643
1a9f6038-dc95-3c44-91df-90ed91ab6050	0.280000	0.828571	0.72	0.680000	0.627143
1a9fda63-96a2-9216-c816-13c890bf35cd	0.970000	0.800000	0.72	0.970000	0.865000
1a9f08a2-476c-a8ea-1e67-f7aa6f2d1f73	0.397143	0.514286	0.28	0.397143	0.397143
1a9fde9d-f4c6-07fd-adf2-865506243cfb	0.970000	0.685714	0.64	0.680000	0.743929

Figure 3: *Extracted data-frame with `NaNs` imputed with mean assessment scores of a student*

Clustering withing each time-step

Using this extracted data-frame we fitted 5 clusters using K-means algorithms at each time-step. The corresponding cluster numbers at a particular time-step were ordinal in nature i.e. cluster 0 represented set of students who scored lower than set of students in cluster 1 and so on. Cluster 4 represented set of students having the best scores.

Also, at every time-step clusters were refitted. It was important to refit new clusters for every assessment because assessments can be inherently different. Using the same fitted clusters for every time-step can be misguiding. For example, suppose assessment 2 was comparatively challenging than assessment 1. Further, suppose that the mean scores for assessments 1 and 2 were 90 and 75 respectively. Now, if a student scores 99 in assessment, he/she will be most likely in cluster 4. But if the same student scored 80 (say this is the highest score in the class) he/she will still be assigned to cluster 3 if we use the same fitted clusters for assessment 1. Considering different score distributions that each assessment can possibly have it is more sensible to refit the clusters.

Figure 4 shows the cluster assignment for the scores at each time-step for the extracted data-frame. *Figure 5* shows the cluster membership path each student follows over the 5 assessments.

	ts0	ts1	ts2	ts3	ts4
s0	3	2	3	4	3
s1	2	2	1	1	1
s2	2	3	3	2	2
s3	3	3	4	3	3
s4	2	2	1	1	1
s5	2	4	3	4	3
s6	1	4	4	3	2
s7	4	4	4	4	4
s8	1	1	0	1	0
s9	4	3	3	3	3
s10	4	4	4	3	3

***Figure 4**: Cluster assignment for the scores at each time-step for the extracted data-frame.*

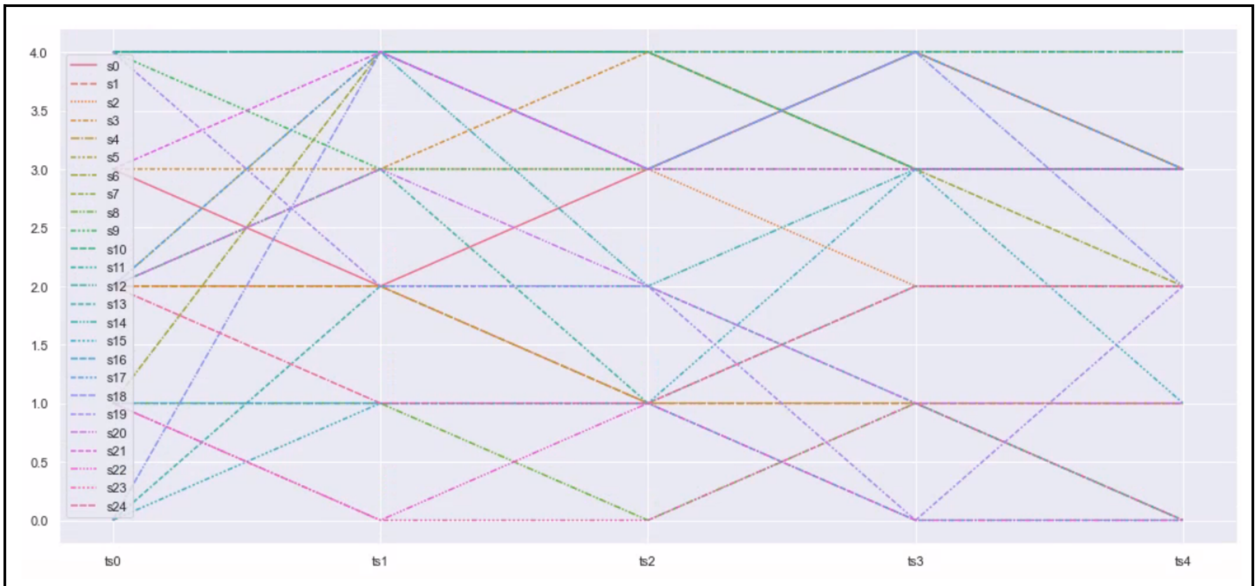
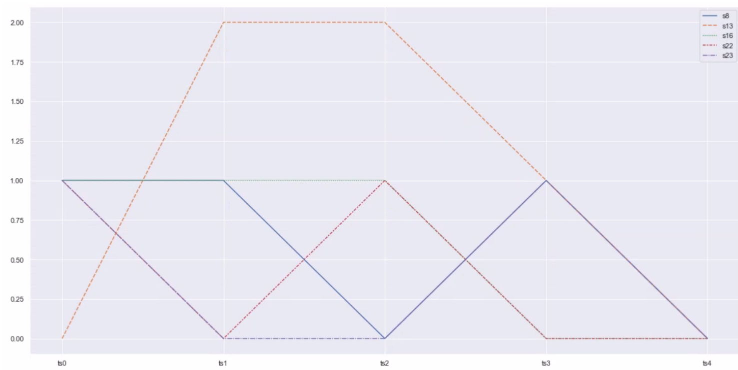


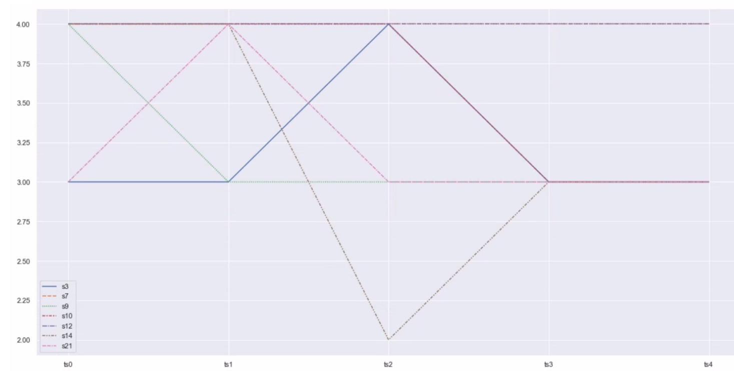
***Figure 5**: Cluster membership path each student follows over the 5 assessments.*

Clustering across time-steps

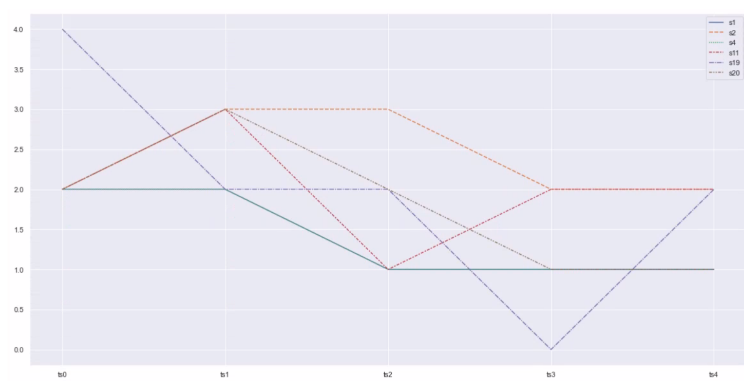
Clearly, from the graph it is hard to find any trend in performance that students have over time. To overcome this problem, we further clustered the paths using K-means clustering algorithm. The idea was to find set of students that have their performance trend close to each other over time. For example, a cluster can represent a set of students which start with high scores at the start of the semester, then have their performance dip in the middle of the semester, but also high scores towards the end. Different clusters can represent different trends. For our case, we found out that the optimum number of clusters was 6 where each cluster represented a distinct trend. *Figure 6* shows the trend for each of the 6 clusters.



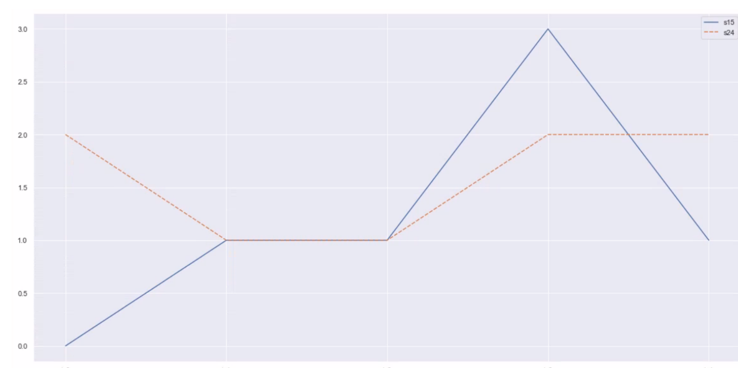
(a) Cluster representing students having low performances in almost all assessments



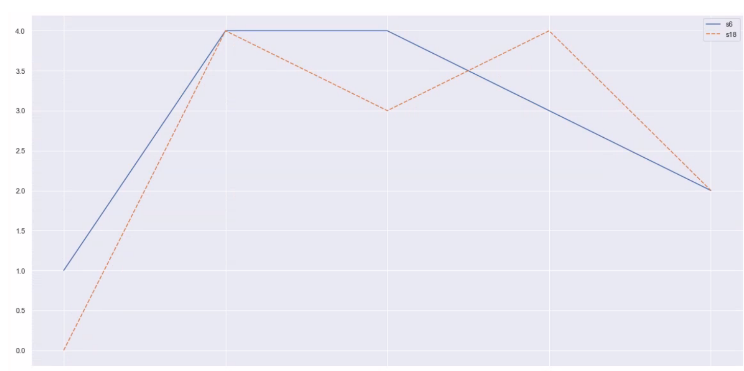
(b) Cluster representing students having high performances in almost all assessments



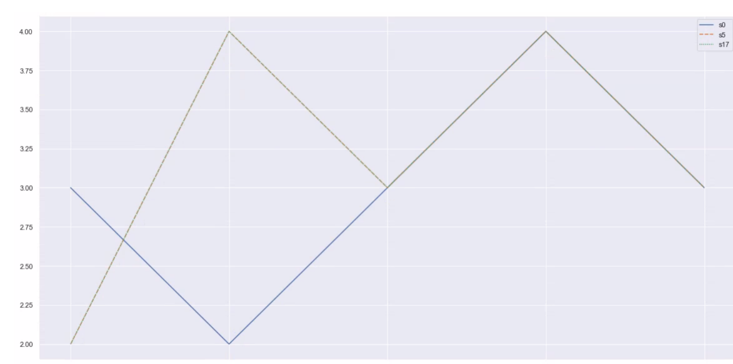
(c) Cluster representing students having a dip in their performance over time



(d) Cluster representing students having a slight rise in their performance over time



(e) Cluster representing students having an initial rise in their performance and a dip at the end of the semester



(f) Cluster representing students having varying performances over time

***Figure 6**: Comparison of trends each cluster represents for student performances over time*