Queen Mary
University of London

Science and Engineering

Understanding and supporting students' learning experience and academic performance using statistical and machine learning

Yulin Tian (E-comm Y3)
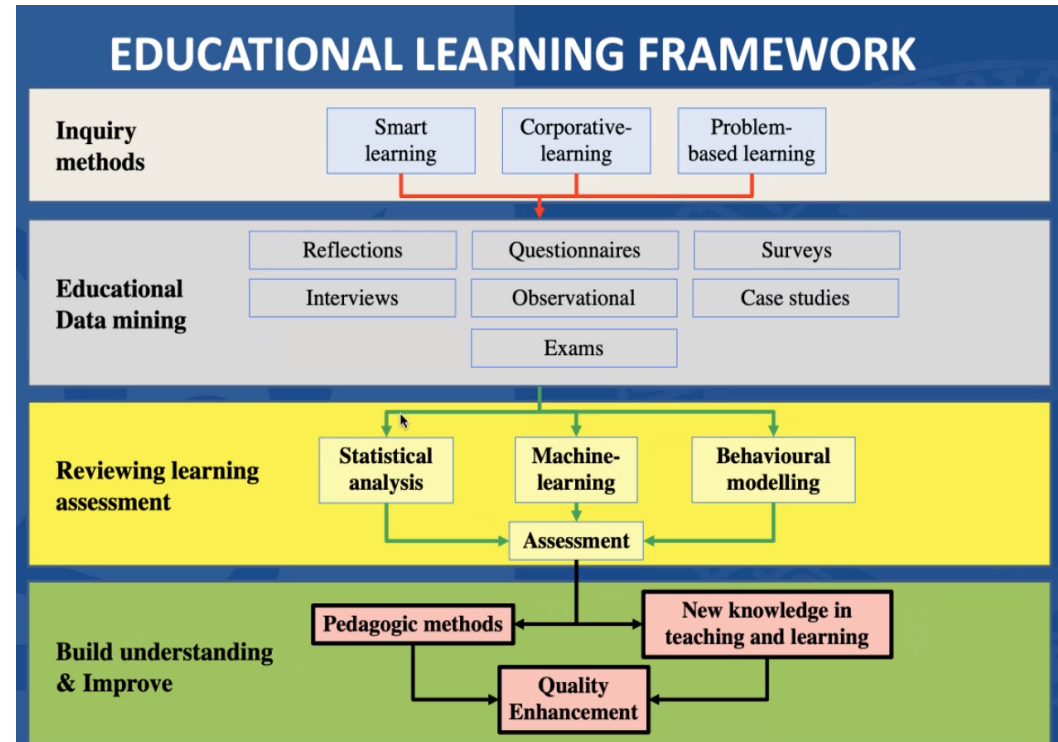
Yuke Zhang (E-comm Y3)

Yumeng Liu (E-comm Y3)

Wenrui Li(IoT Y3)

Yunkai Li (IoT Y3)

Project Leader: Dr Ethan Lau

Introduction

- ➢ Basic information of project
  - • Learning behavior under online environment
  - • Using of statistical analysis and machine learning
  - • Website design
- ➢ Framework of the project

EDUCATIONAL LEARNING FRAMEWORK

Inquiry methods

| Smart learning | Corporative-learning | Problem-based learning |

Educational Data mining

Reflections	Questionnaires	Surveys
Interviews	Observational	Case studies
	Exams	

Reviewing learning assessment

| Statistical analysis | Machine-learning | Behavioural modelling |

Assessment

Build understanding & Improve

Pedagogic methods — New knowledge in teaching and learning

Quality Enhancement

Part 1 -Data Analysis & Visualization

- Integrated & Sorted Data

21_22 Original dataset

- ✓ CWK.xlsx
- ✓ Distribution of exam scores.xlsx
- ✓ Grades.xlsx
- ✓ Results.csv

Integrated dataset

- ➤ Personal Information
  - BUPT Student ID & ID number
- ➤ Academic Performance
  - Grades of each assignments
  - Class Test
  - Exam & Each question
  - Overall QMUL & BUPT Result
- ➤ Information related to watching videos
  - Engagement
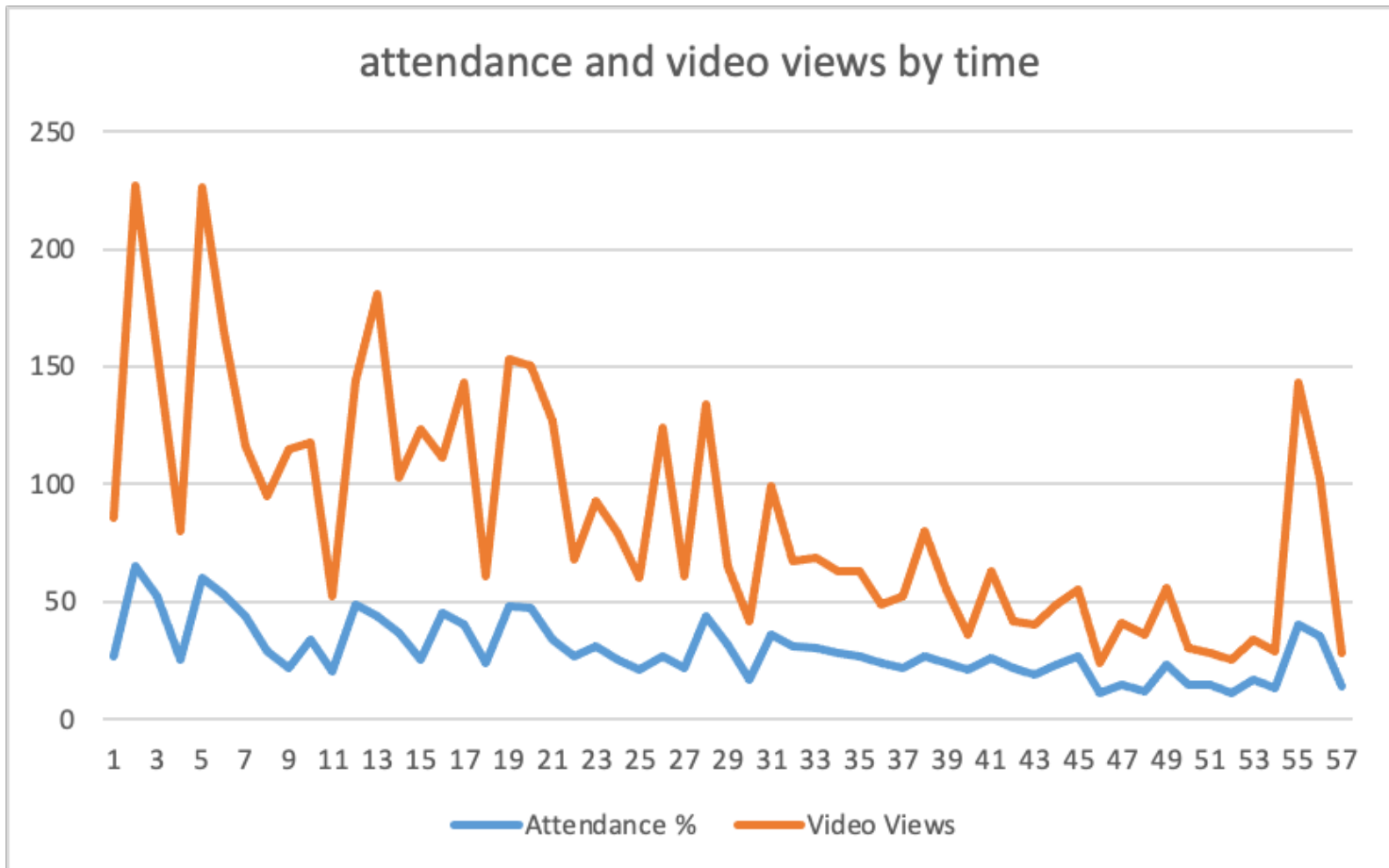 - Attendance
  - Video views

- Classified Dataset

Overall Grade
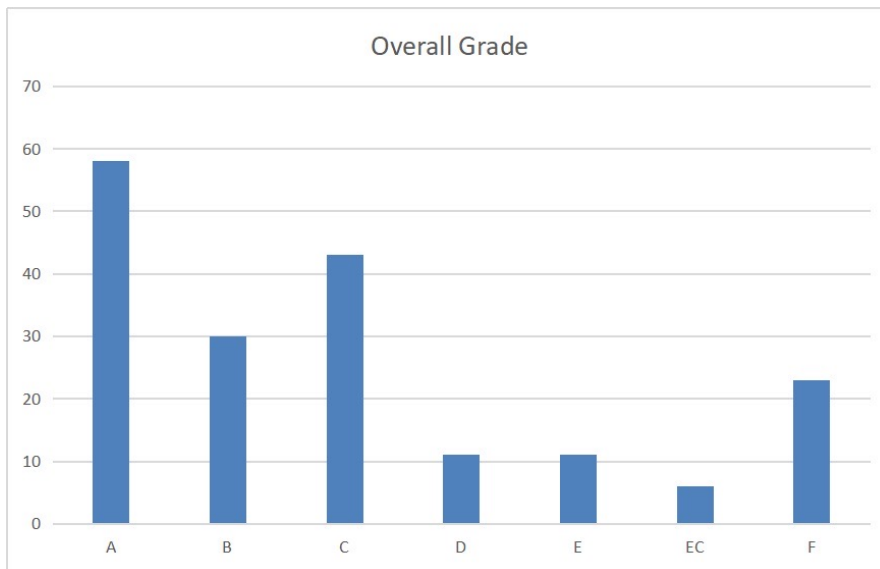
Data Analysis & Visualization

- Positive Correlation between Exam and Class Test

correlation between exam and class test
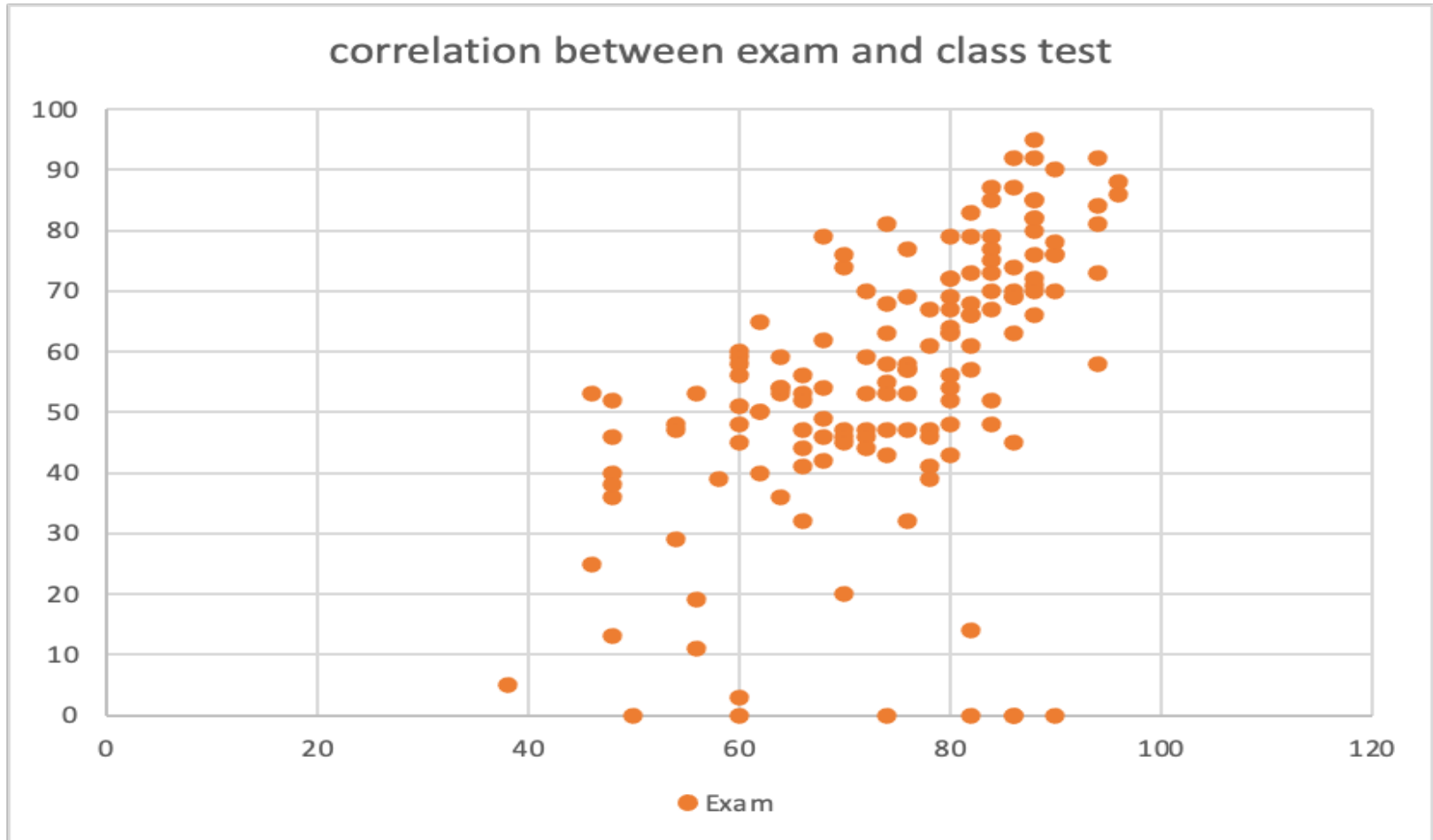
Data Analysis & Visualization

- Attendance & video views

5

Data Analysis & Visualization

- Preliminary conclusions

  For 21_22 Students

- Shows little correlation with the scores level
  - Total engagement
  - Attendance
  - The number of video views - low in general
  - Weighted engagement (discrete distribution)
  - Video view% (discrete distribution)

- Some correlation between class tests and scores levels of students
- Some correlation with the coursework and exam grades
- Positive correlation between exam and class test scores
- The attendence and video views decreased over time

Part 2 - Affinity Analysis

- Use Apriori algorithm to find the **specific video's relationship**

First step

✓ Find frequent itemsets in the data using the Apriori algorithm

Second step

➢ Create association rules from those itemsets

Affinity Analysis

There are the rules between the videos of the **training set**

```
Rule #1
Rule: If a student watches frozenset({'20210917 — IoT_G1_G2 Recorded — JMS — Part B'}) they will also watch 2021091
7 — IoT_G1_G2 Recorded — JMS — Part A
 — Confidence: 1.000

Rule #2
Rule: If a student watches frozenset({'2021014 — IoT_G1_G2 Recorded — FurtherProgramming_Spring_Framework — Part A'
, '20210917 — IoT_G1_G2 Recorded — JMS — Part B'}) they will also watch 20210917 — IoT_G1_G2 Recorded — JMS — Part
A
 — Confidence: 1.000

Rule #3
Rule: If a student watches frozenset({'20210930 — IoT_G1_G2 Recorded — Spring Framework — Part B', '2021018 — IoT_G
1_G2 Recorded — Threads_and_Concurrency — Part A'}) they will also watch 2021018 — IoT_G1_G2 Recorded — Threads_and
_Concurrency — Part B
 — Confidence: 1.000

Rule #4
Rule: If a student watches frozenset({'20210930 — IoT_G1_G2 Recorded — Spring Framework — Part B', '2021018 — IoT_G
1_G2 Recorded — Threads_and_Concurrency — Part B'}) they will also watch 2021018 — IoT_G1_G2 Recorded — Threads_and
_Concurrency — Part A
 — Confidence: 1.000

Rule #5
Rule: If a student watches frozenset({'2021018 — IoT_G1_G2 Recorded — Threads_and_Concurrency — Part B', '20210920
— IoT_G1_G2 Recorded — JSP — Part B'}) they will also watch 2021018 — IoT_G1_G2 Recorded — Threads_and_Concurrency
— Part A
 — Confidence: 1.000
```

Affinity Analysis

There are the rules between the videos of the **testing set**

```
Rule #1
Rule: If a student watches frozenset({'20210917 – IoT_G1_G2 Recorded – JMS – Part B'}) they will also watch 2021091
7 – IoT_G1_G2 Recorded – JMS – Part A
 – Train Confidence: 1.000
 – Test Confidence: 0.964

Rule #2
Rule: If a student watches frozenset({'2021014 – IoT_G1_G2 Recorded – FurtherProgramming_Spring_Framework – Part A'
, '20210917 – IoT_G1_G2 Recorded – JMS – Part B'}) they will also watch 20210917 – IoT_G1_G2 Recorded – JMS – Part
A
 – Train Confidence: 1.000
 – Test Confidence: 0.947

Rule #3
Rule: If a student watches frozenset({'20210930 – IoT_G1_G2 Recorded – Spring Framework – Part B', '2021018 – IoT_G
1_G2 Recorded – Threads_and_Concurrency – Part A'}) they will also watch 2021018 – IoT_G1_G2 Recorded – Threads_and
_Concurrency – Part B
 – Train Confidence: 1.000
 – Test Confidence: 0.909

Rule #4
Rule: If a student watches frozenset({'20210930 – IoT_G1_G2 Recorded – Spring Framework – Part B', '2021018 – IoT_G
1_G2 Recorded – Threads_and_Concurrency – Part B'}) they will also watch 2021018 – IoT_G1_G2 Recorded – Threads_and
_Concurrency – Part A
 – Train Confidence: 1.000
 – Test Confidence: 1.000

Rule #5
Rule: If a student watches frozenset({'2021018 – IoT_G1_G2 Recorded – Threads_and_Concurrency – Part B', '20210920
– IoT_G1_G2 Recorded – JSP – Part B'}) they will also watch 2021018 – IoT_G1_G2 Recorded – Threads_and_Concurrency
– Part A
 – Train Confidence: 1.000
 – Test Confidence: 1.000
```
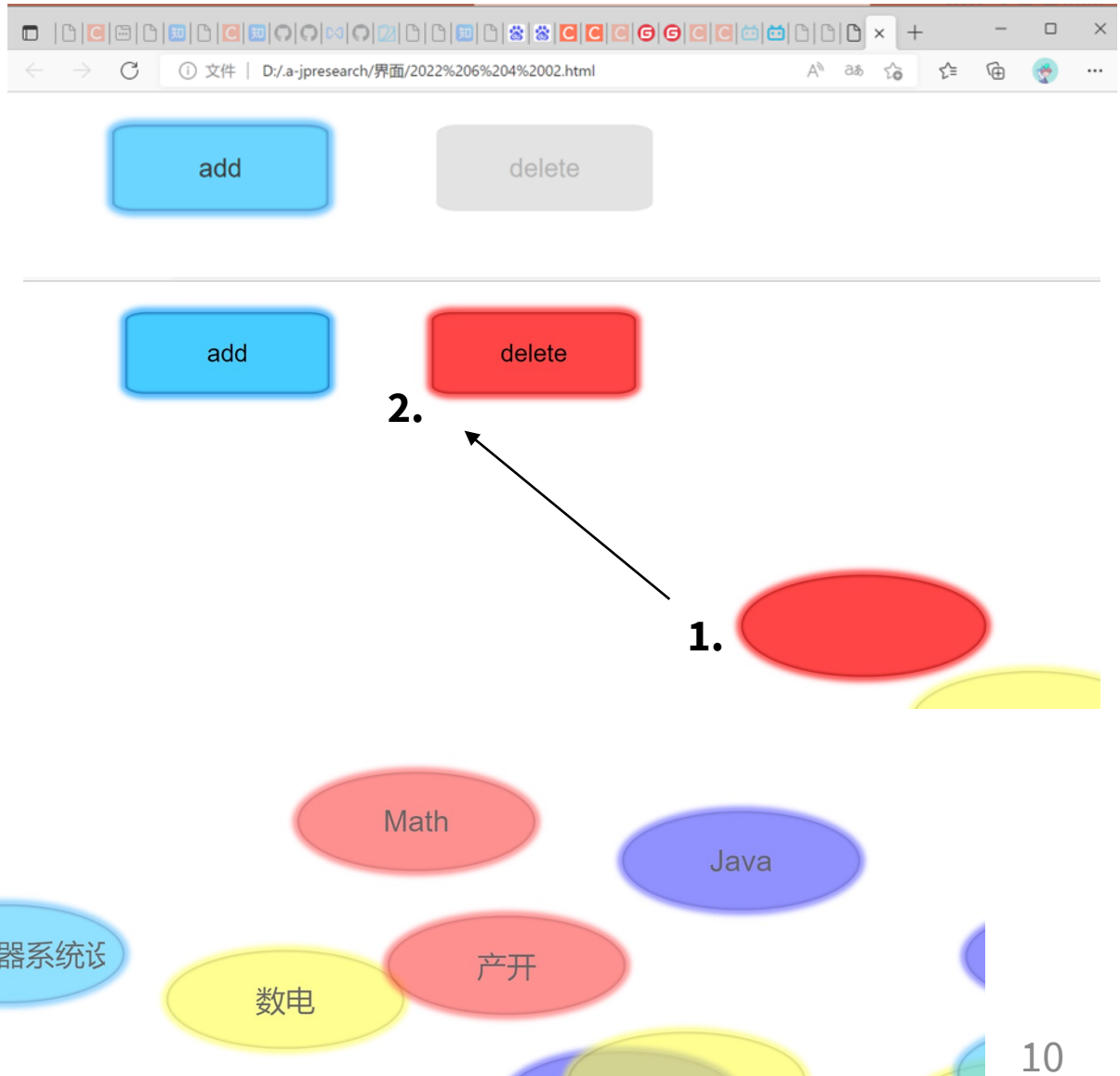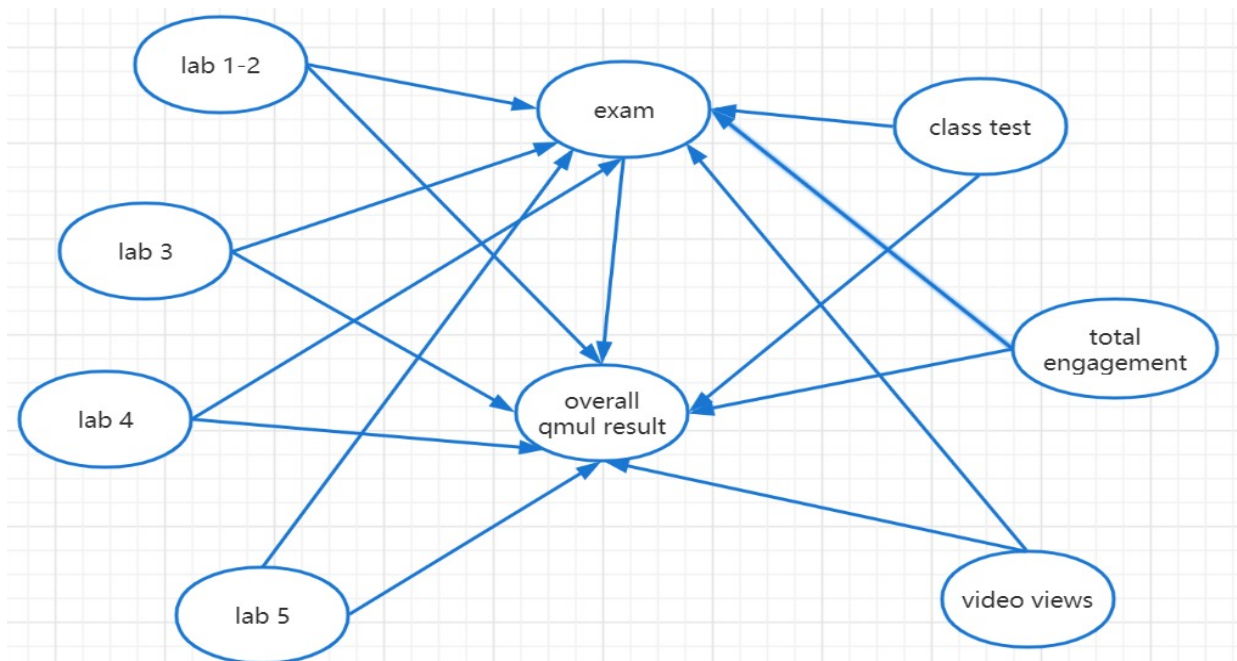
Part 3 - Website UI Design

- HTML Page
- Functions：
  - Add
  - Delete
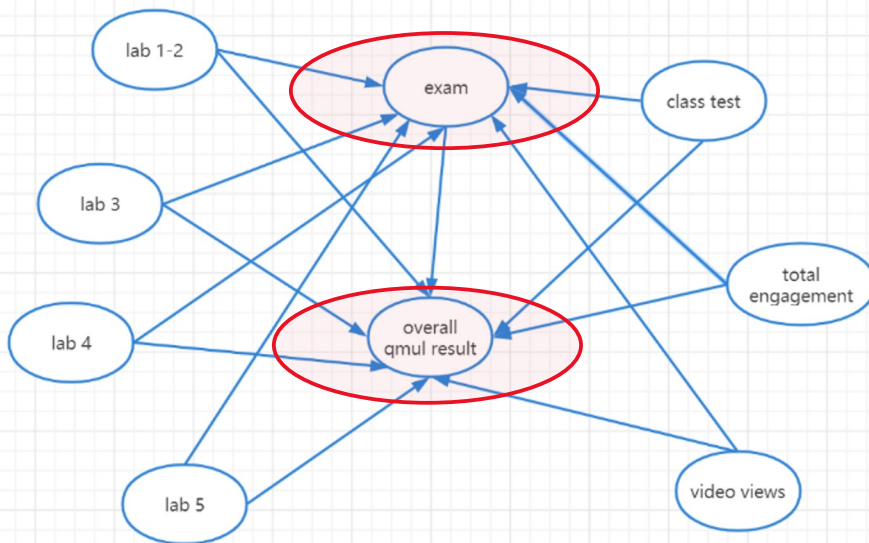 - Write
  - Drag

Part 3 - Bayesian Network

- DAG
- Node: random variables
- Edge: relationships between nodes
- Conditional Probability

$$p(x) = \prod_{i \in I} p\left(x_i \mid x_{\mathbf{pa}(i)}\right)$$

Problem Encountered

- Continuous-> Discrete
- Complexity of Network

Immediate and future work

- Construct multi-dimensional networks for relationship description and prediction
    - interaction network
    - Bayesian network

- Utilize multi-modality information to understand student behaviors
    - evaluation metrics like video view, exam result etc
    - statistical characteristics

- Develop strategies for learning quality enhancement in new-normal environment
    - personal estimation / group orientation
    - user-friendly interface