Job market research for data analyst positions in the US

Xiaotong Liu

Background

Data science, analytics, AI, big data are becoming widely used in many fields, which leads to the ever-increasing demand for data analysts, data scientists, ML engineers, managers of analytics and other data professionals. Due to that, data science education is now a hot topic for educators and entrepreneurs. To figure out the skills that are in demand in the job market from job vacancies posted on http://indeed.com web-portal, our project collected data using web scraping on indeed to get the skill requirement for the data analyst position in the US. After the data quality check and data cleaning, we performed exploratory data analysis and feature engineering; Two machine learning clustering algorithms were implemented. At the end of the report, the data analysis findings were interpreted.

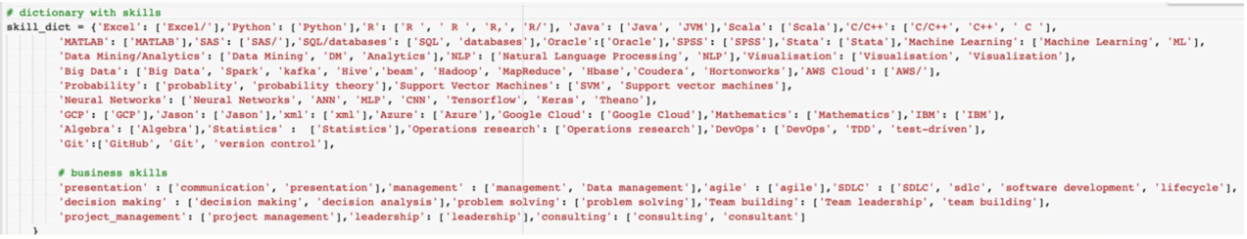
Data collection and cleaning

In this report, we will focus on the data analyst position in the US. First, we do web scraping on indeed for the job posting in the US for data analyst. After web scaping, we use `*drop_duplicates()*` for data cleaning and get 1097 job postings for data analyst positions in the US. Our dataset contains job tittle, company, job location, rating, salary, job posting links and job description.

Exploratory data analysis and feature engineering

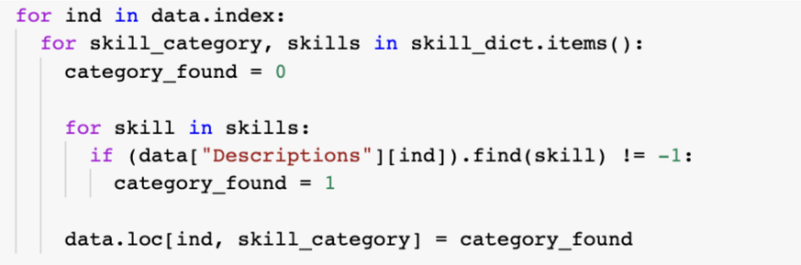
First, we create a dictionary `skills_keywords_dict`, in which contains common technical skills and business skills data analyst needed.

Here since some keywords like `R` may appear in some other words, so we specify different situation for the appearance of the keywords and avoid mis-extracting. The dictionary is listed as following:



```
# dictionary with skills
skill_dict = {'Excel': ['Excel/'],'Python': ['Python'],'R': ['R ', ' R ', 'R,', 'R/'], 'Java': ['Java', 'JVM'],'Scala': ['Scala'],'C/C++': ['C/C++', 'C++', ' C '],
        'MATLAB': ['MATLAB'],'SAS': ['SAS/'],'SQL/databases': ['SQL', 'databases'],'Oracle':['Oracle'],'SPSS': ['SPSS'],'Stata': ['Stata'],'Machine Learning': ['Machine Learning', 'ML'],
        'Data Mining/Analytics': ['Data Mining', 'DM', 'Analytics'],'NLP': ['Natural Language Processing', 'NLP'],'Visualisation': ['Visualisation', 'Visualization'],
        'Big Data': ['Big Data', 'Spark', 'kafka', 'Hive','beam', 'Hadoop', 'MapReduce', 'Hbase','Coudera', 'Hortonworks'],'AWS Cloud': ['AWS/'],
        'Probability': ['probablity', 'probability theory'],'Support Vector Machines': ['SVM', 'Support vector machines'],
        'Neural Networks': ['Neural Networks', 'ANN', 'MLP', 'CNN', 'Tensorflow', 'Keras', 'Theano'],
        'GCP': ['GCP'],'Jason': ['Jason'],'xml': ['xml'],'Azure': ['Azure'],'Google Cloud': ['Google Cloud'],'Mathematics': ['Mathematics'],'IBM': ['IBM'],
        'Algebra': ['Algebra'],'Statistics' :  ['Statistics'],'Operations research': ['Operations research'],'DevOps': ['DevOps', 'TDD', 'test-driven'],
        'Git':['GitHub', 'Git', 'version control'],

        # business skills
        'presentation' : ['communication', 'presentation'],'management' : ['management', 'Data management'],'agile' : ['agile'],'SDLC' : ['SDLC', 'sdlc', 'software development', 'lifecycle'],
        'decision making' : ['decision making', 'decision analysis'],'problem solving': ['problem solving'],'Team building': ['Team leadership', 'team building'],
        'project_management': ['project management'],'leadership': ['leadership'],'consulting': ['consulting', 'consultant']
    }
```

The larger the number of the skill appears in the job description, the skill is assumed to be more important, so here we first convert the job description column into dummy variable for each skill, with 0 represents the skill does not appear in the job description and 1 represents the skill appear in the job description.

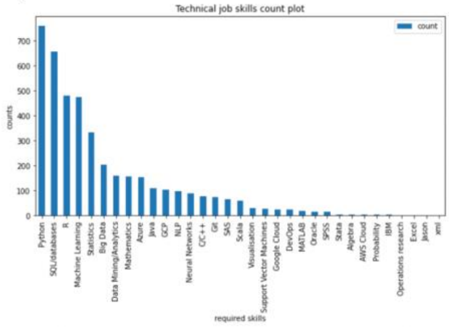


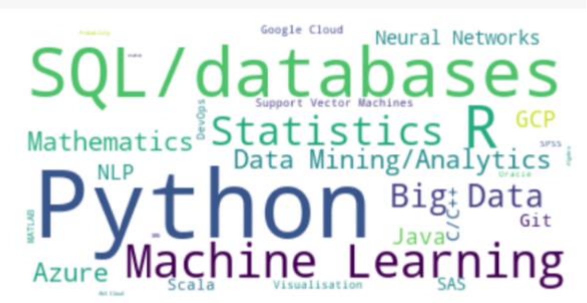
```
for ind in data.index:
  for skill_category, skills in skill_dict.items():
    category_found = 0

    for skill in skills:
      if (data["Descriptions"][ind]).find(skill) != -1:
        category_found = 1

    data.loc[ind, skill_category] = category_found
```

After that, we count the number of 1 for each skill and add one row to our dataframe called *count*. Here the value of count represents the number of appearances for each skill. Then we visualize the word cloud for technical/hard skills, business/soft skills, and all skills respectively.

- Word cloud of technical/hard skills

First, we create a dataframe `*data_tech*`, in which contains all the data for technical skills. Then we first use bar chart to have a rough understanding the overall count (ranking) for each skill.
Now we will plot word cloud for data visualization, which could help the reader of the report have direct understanding of the importance for tech/hard skills.

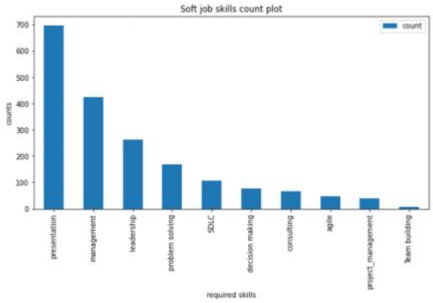


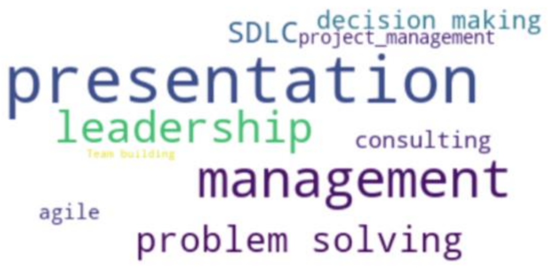


From the word cloud above, we could notice directly that Python, SQL/databases, R, Machine Learning, Statistics are relative important compared to other skills.

- Word cloud of business/soft skills

In the same way, now we will plot the word cloud for business/soft skills.

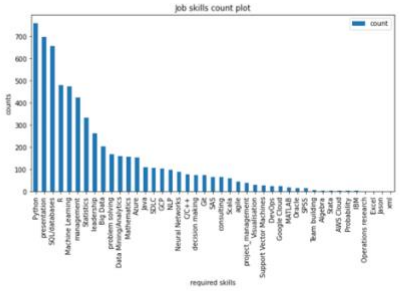


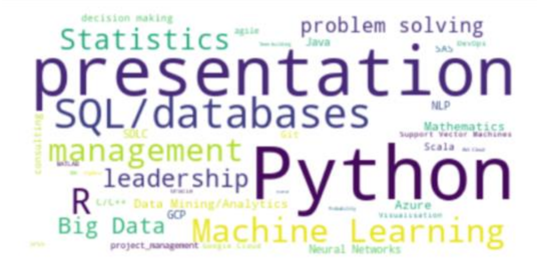


From the word cloud above, we could notice directly that presentation, management, leadership, problem solving are relative important compared to other soft skills.

- Word cloud of all skills

Now we will plot the word cloud of all skills using the same way as before.





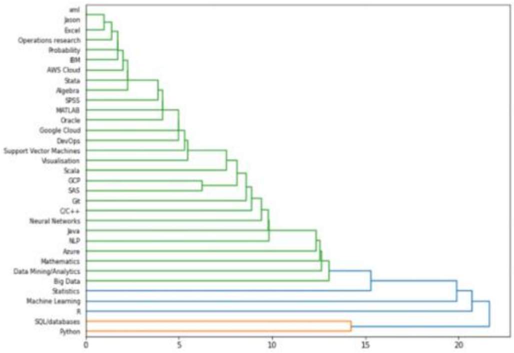
From the word cloud above, we could notice directly that Python, presentation, SQL/database, R, Machine Learning are relative important compared to other skills.

Hierarchical clustering implementation

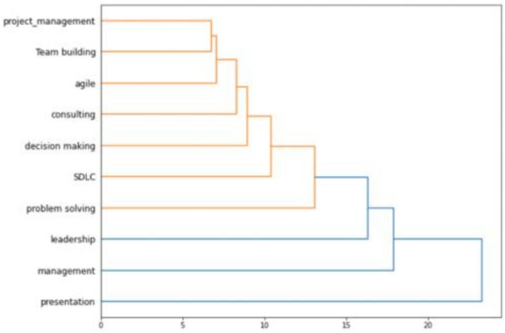
First, we use hierarchical clustering. Initially each data point is considered as an individual cluster. At each iteration, the similar clusters merge with other clusters until all the clusters are formed. Here the centroid-linkage is used to measure the distance and we get the following result. In order to have a better understanding of the relationship between each skill within different kinds of skills (i.e., tech and soft), we first doing hierarchical clustering for tech and soft separately.

1. Hierarchical clustering for technical/hard skills

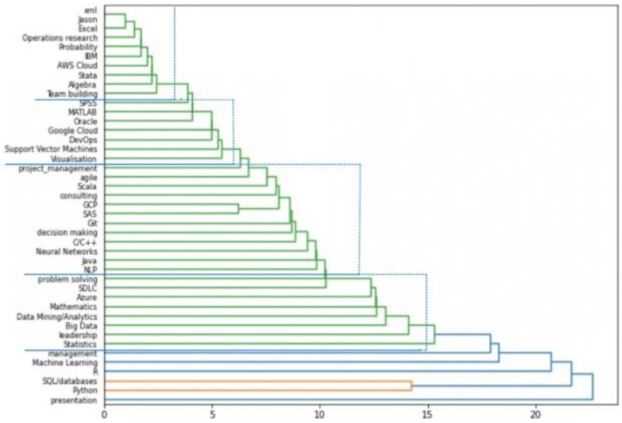
First, we implement the hierarchical clustering for technical/hard skills and got the plot:



2. Hierarchical clustering for business/soft skills



3. Hierarchical clustering for all skills



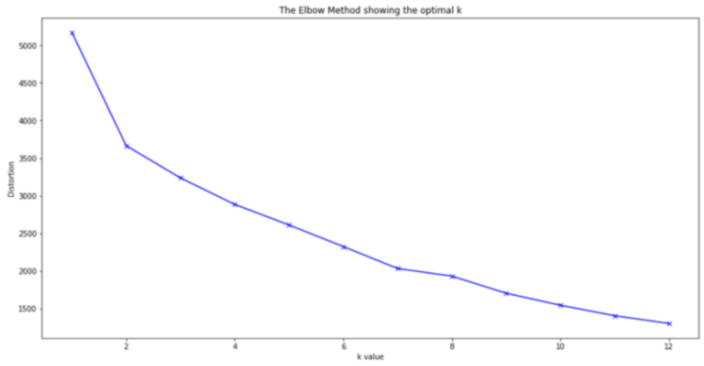
The number of clusters in hierarchical clustering in this report is decided based on the previous experience about data analysis courses and the relevance of each skill. From hierarchical plot above, for hierarchical clustering we have 6 clusters:

1. We divide clusters in green into four clusters and shown in the graph above
2. SQL and Python as one cluster
3. Management, machine learning, R, presentation as one cluster

K-Mean clustering implementation

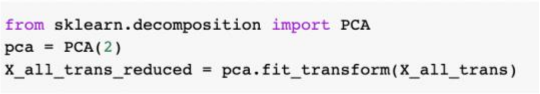
Here we use K-Means clustering to cluster each skill. Since we have dimension 1097 (number of sample) * 43 (number of skill) for our dataset, so we first transpose the dataset so that we could cluster skills rather than samples.

To determine the number of clusters, we use elbow method to help us. In elbow method, we would choose a value of k where the SSE begins to flatten out and we see an inflection point. If we choose k from 8-12, we may not have the right k, so in order to have an accurate value for number of cluster, we choose k from 1 to 12 and got the elbow plot below:



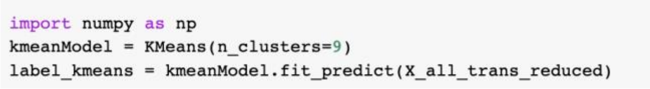
From the plot, since we need to design a sequence of 8-12 courses, so we choose we choose k = 9, at which the slope of the line changed. Next, we will implement K-Mean model with 9 clusters.

Here we first apply PCA before K-Means clustering. PCA improves the clustering results in practice (noise reduction) and we want to keep in 2D, so we apply PCA (2) to our dataset.



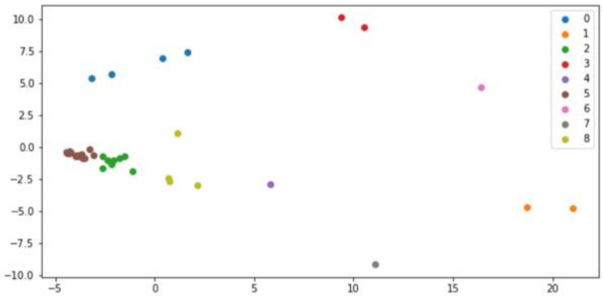
```
from sklearn.decomposition import PCA
pca = PCA(2)
X_all_trans_reduced = pca.fit_transform(X_all_trans)
```

Then we build K-Mean clustering with 9 clusters and fit the model to each skill.

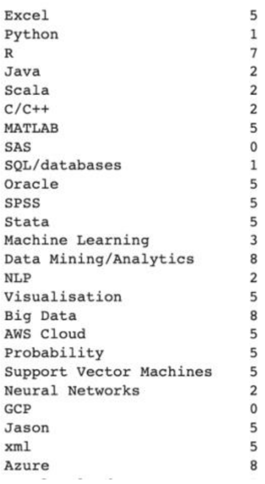


```
import numpy as np
kmeanModel = KMeans(n_clusters=9)
label_kmeans = kmeanModel.fit_predict(X_all_trans_reduced)
```

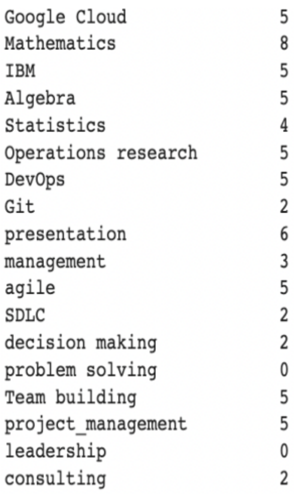
And use scatterplot to visualize the cluster result.



To know the clustering in detail, we combine labels with skills and got the results.



```
Excel                        5
Python                       1
R                            7
Java                         2
Scala                        2
C/C++                        2
MATLAB                       5
SAS                          0
SQL/databases                1
Oracle                       5
SPSS                         5
Stata                        5
Machine Learning             3
Data Mining/Analytics        8
NLP                          2
Visualisation                5
Big Data                     8
AWS Cloud                    5
Probability                  5
Support Vector Machines      5
Neural Networks              2
GCP                          0
Jason                        5
xml                          5
Azure                        8
```



```
Google Cloud             5
Mathematics              8
IBM                      5
Algebra                  5
Statistics               4
Operations research      5
DevOps                   5
Git                      2
presentation             6
management               3
agile                    5
SDLC                     2
decision making          2
problem solving          0
Team building            5
project_management       5
leadership               0
consulting               2
```

From the elbow plot above, when we have 9 clusters, the SSE begins to flatten out and we see an inflection point for K-Mean clustering, so I choose 9 clusters, with each cluster and its related skills as following:

1. Management
2. Big Data
3. Python and SQL/databases
4. SAS, GCP, problem solving and leadership

5. presentation
6. Statistics
7. R
8. Machine Learning
9. Excel, Java, Scala, C/C++, MATLAB, Oracle, SPSS, Stata, Data Mining/Analytics, NLP, Visualization, AWS Cloud, Probability, Support Vector Machines, Neural Networks, Jason, xml, Azure, Google Cloud, Mathematics, IBM, Algebra, Operations research, DevOps, Git, agile, SDLC, decision making, Team building, project management, consulting.

Interpretation of results, discussion and final course curriculum

- Course curriculum for K-mean clustering

From the result in part 4, we have 9 clusters. We design our course curriculum as following:

From the 9-th category, based on our knowledge to the course, we first choose skills which related to mathematical or data analysis knowledge and computational skills as courses, then we have courses: Data Mining/Analytics, NLP, Data Visualization, Introduction to Probability, Introduction to Neural Networks and Operations research.

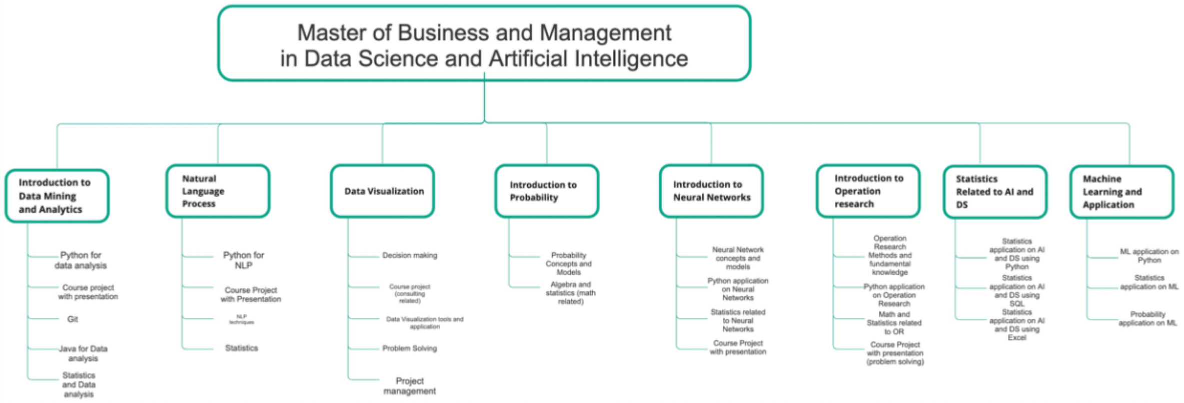
Then based on the scatterplot, we find skills close to 9th category, and we have result as following:

- Data Mining/Analytics: Python for data analysis, Course Project with presentation, Git, Java for data analysis, statistics, and data analysis.
- NLP: Python for NLP, Course project with presentation, statistics, NLP techniques
- Data Visualization: Decision making, Course project (consulting related), Data visualization tools and application, Problem solving, Project Management.
- Introduction to Probability: Probability concepts and models, Algebra and statistics (math related).
- Introduction to Neural Networks: Neural Network concepts and models, Python application on neural network, statistics related to neural network, course project with presentation.
- Operations research: Operation Research Methods and fundamental knowledge, Python application on Operation Research, Math and Statistics related to OR, Course Project with presentation (problem solving)

Similarly, from the other category, we have course:

- Statistics Related to AI and DS: Statistics application on AI and DS using Python, Statistics application on AI and DS using AI, Statistics application on AI and DS using Excel.

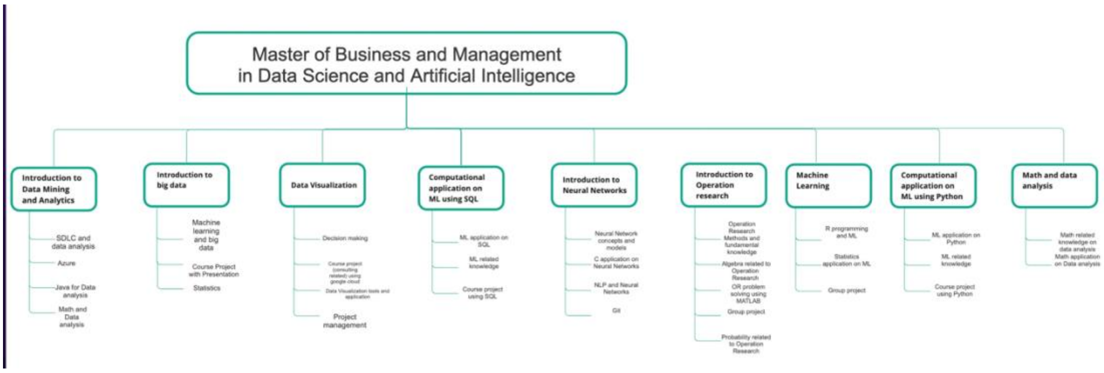
- Machine Learning and Application: ML application on Python, Statistics application on ML, Probability application on ML.


Master of Business and Management
in Data Science and Artificial Intelligence
Introduction to Data Mining and Analytics
Python for data analysis
Course project with presentation
Git
Java for Data analysis
Statistics and Data analysis
Natural Language Process
Python for NLP
Course Project with Presentation
NLP techniques
Statistics
Data Visualization
Decision making
Course project (consulting related)
Data Visualization tools and application
Problem Solving
Project management
Introduction to Probability
Probability Concepts and Models
Algebra and statistics (math related)
Introduction to Neural Networks
Neural Network concepts and models
Python application on Neural Networks
Statistics related to Neural Networks
Course Project with presentation
Introduction to Operation research
Operation Research Methods and fundamental knowledge
Python application on Operation Research
Math and Statistics related to OR
Course Project with presentation (problem solving)
Statistics Related to AI and DS
Statistics application on AI and DS using Python
Statistics application on AI and DS using SQL
Statistics application on AI and DS using Excel
Machine Learning and Application
ML application on Python
Statistics application on ML
Probability application on ML

● Course curriculum for hierarchical clustering

From the result in part 3, we have 6 clusters. We design our course curriculum as following:

1. For the skills with green line (4 clusters), based on study experience and characteristic of subject, we want to divide them into more specific skill categories, so we divide them into 6 courses: Introduction to Operation Research, Data Visualization, Introduction to Neural Networks, Math and data analysis, Introduction to Data Mining and Analytics, Introduction to big data.
2. Similarly, for the other two category, we have three courses: Machine Learning and Computational application on ML using SQL and Computational application on ML using Python. Since SQL and Python are relatively large topic, so we divide them into two courses.


Master of Business and Management
in Data Science and Artificial Intelligence
Introduction to Data Mining and Analytics
SDLC and data analysis
Azure
Java for Data analysis
Math and Data analysis
Introduction to big data
Machine learning and big data
Course Project with Presentation
Statistics
Data Visualization
Decision making
Course project (consulting related) using google cloud
Data Visualization tools and application
Project management
Computational application on ML using SQL
ML application on SQL
ML related knowledge
Course project using SQL
Introduction to Neural Networks
Neural Network concepts and models
C application on Neural Networks
NLP and Neural Networks
Git
Introduction to Operation research
Operation Research Methods and fundamental knowledge
Algebra related to Operation Research
OR problem solving using MATLAB
Group project
Probability related to Operation Research
Machine Learning
R programming and ML
Statistics application on ML
Group project
Computational application on ML using Python
ML application on Python
ML related knowledge
Course project using Python
Math and data analysis
Math related knowledge on data analysis
Math application on Data analysis

Comparison: There are 8 courses in the course curriculum for K-mean clustering and there are 9 courses in the course curriculum for hierarchical clustering. Based on the course content, the coverage for courses designed using K-mean clustering is larger and includes more topics. I choose course curriculum for K-mean clustering as the final course curriculum. There is a creative way for course design, Python and SQL are necessary skills for data analyst, however, some students may not have enough coding background for the course curriculum designed in the report. One way could solve this problem is choose two fundamental coding courses in the later curriculum——

computational application on ML using SQL and Pythons as pre-requisite for ML and application and design a coding test for it. If a student could pass the test, then he could take the course without taking these two.