

**DSO 570 Final Project Report**

**Marshall Class Scheduling**



May 12, 2020

**Session: 12:30PM**

Team Members

Chutong Yan, Ji Jie, Astha Srivastava, Rory Wang, Jayant Maheshwari

1. Executive Summary

The USC Marshall School of Business has 7 departments, 22 academic programs, and enrollment of around 5000 undergraduate students and 1000 graduate students. With multiple programs, priorities and preferences, it is difficult for Institutional Research and Academic Administration to create a classroom schedule without scrambling and headaches. Hence, the team was hired by Marshall to investigate the possibility of applying optimization to improve the efficiency and transparency of the current system and to rigorously quantify the potential gains.

In the process of data exploration based on historical schedule, the team identified three major pain points:

* **Time-consuming scheduling methodology**
* **Inappropriate utilization of classroom space**
* **Professors’ preferences not satisfied**. Specifically, the team investigated two general trend regarding professors’ preference: 1) Avoid working more than two days a week 2) Desire back-to-back classes

In order to address the opportunities for improvement, the team proposed an optimization tool prototype which intakes the readily available time slots, classroom and class information, and outputs the schedule for all classes and a summary of the schedule performance. To estimate the potential gain from the optimization, the team applied the tool using sample data based on classes in the 2019 Spring term, with 533 classes, 38 classrooms and 241 timeslots. As a result, the prototype generated a new free-from-conflict schedule within 40 minutes, which pushes the number of professors who have to work more than two days a week to zero, increases the average utilization rate to 89.58%, and increases the proportion of professors who are allocated at least one back-to-back class by 17.5%.

Based on the results, the team offered Shannon and her team with a series of final recommendations:

* **Pilot the optimization tool with a small section of classes to gain confidence**. For example, utilize the model to schedule only graduate classes or MBA classes.
* **Select the weight of scheduling features by sensitive analysis**. Scheduling features include average classroom utilization rate, number of professors who are allocated at least one back-to-back class and number of professors who have to work more than two days a week. By default the weight for the features above are 1, 1 and 0.2.
* **Adopt the output of the optimization tool as a scheduling start point**, **then address special cases manually**. For example, very few graduate classes with length more than 3 hours are not considered by the model and need adjustment by hand.
* **Partition the classes and run parallely to save runtime**. Currently, allocating regular spring term classes takes the model 25 - 40 minutes. To save the runtime, it’s suggested to split the input courses classrooms and time slots into 2 - 4 non-overlapping parts, run each part parallely, and finally concat the outputs to create the full schedule.

2. Opportunity for Improvement

Based on exploration of 2015 - 2019 schedules and faculty survey results, the team identified and focused on three major opportunities for improvement, which are explained in detail in 2.1, 2.2 and 2.3.

## 2.1 Time Consuming Scheduling Methodology

As addressed by Shannon and Hal from the Marshall Office of Finance and Administration, the scheduling work of allocating courses and classrooms at USC Marshall has to begin almost one year before the semester begins. For spring 2019, the initial allocation is made at the beginning of spring 2018.

The first phase of the allocation work begins with Marshall Office giving out assigned time slots to each department in Marshall business school, then each department coordinator is responsible to populate the slots with courses as well as assign teaching duties to individual professors. During this phase, although the allocation is based hugely on historical schedules, the task still consists of making changes to new courses or new sections of existing courses, and taking professor preferences into consideration, and all work mentioned above are done on an ad-hoc basis, which is inefficient, labor costly and time consuming. This situation has resulted in a lot of last minute scrambling and headaches for certain departments in the past. In rare cases, the classroom may still be unassigned at the time students choose classes.

Since improving the efficiency and complexity of the scheduling process could not only decrease the workload of the administration office, but also beneficial to the student as there would be less occurrences in which students have to make a course selection with unassigned slot, it’s important to build a reusable system which would assist in the scheduling process with efficient and correctness, and take the burden of course assignment task off the office staff’s chest.

Based on the methodology, the team is devoted to building an efficient schedule recommending model, which would generate a recommended schedule in a timely manner, and assist in increasing the efficiency of the administration office.

## 2.2 Inappropriate Utilization of Classroom Space

The capacity of Marshall classrooms and seats offered for each class ranges widely from 15 to more than 200. To ensure efficient usage of classroom space, it’s critical to match the size of class and the capacity of the room.

To evaluate the extent of space utilization, the team create three metrics:

1. **Minimum utilization rate in each term,** which quantifies the extreme cases of classroom usage
2. **Average utilization rate in each term**, which quantifies the overall classroom usage. Inevitably, the normal average is somewhat biased as classrooms of different sizes are given the same weight. However, for the purpose of simplicity, the team decided to adopt ordinary average statistics as the evaluation metric.
3. **Percentage/number of classes utilized less than ¾ of classroom space**. Based on research, it is said that utilization rate between 70% to 80% is commonly considered as good (classroom and class laboratory utilization Analysis, Dec 2014), thus the team decided to take 75% as the threshold for *efficient* utilization.

Looking across all terms regarding the first metric, the team discovered that sometimes, the utilization rate can be as low as 3.7 % (Table 2.1), which is undoubtedly a huge waste of space. More specifically, in term 20191, ACCT - 574, a class only offering 4 seats, was assigned to room JKP202, a room that can accomodate 54 people. Considering the fact that there were available smaller rooms, such as ACC 312 with capacity of 20, this extreme case indicates the inappropriateness in matching class and room size.

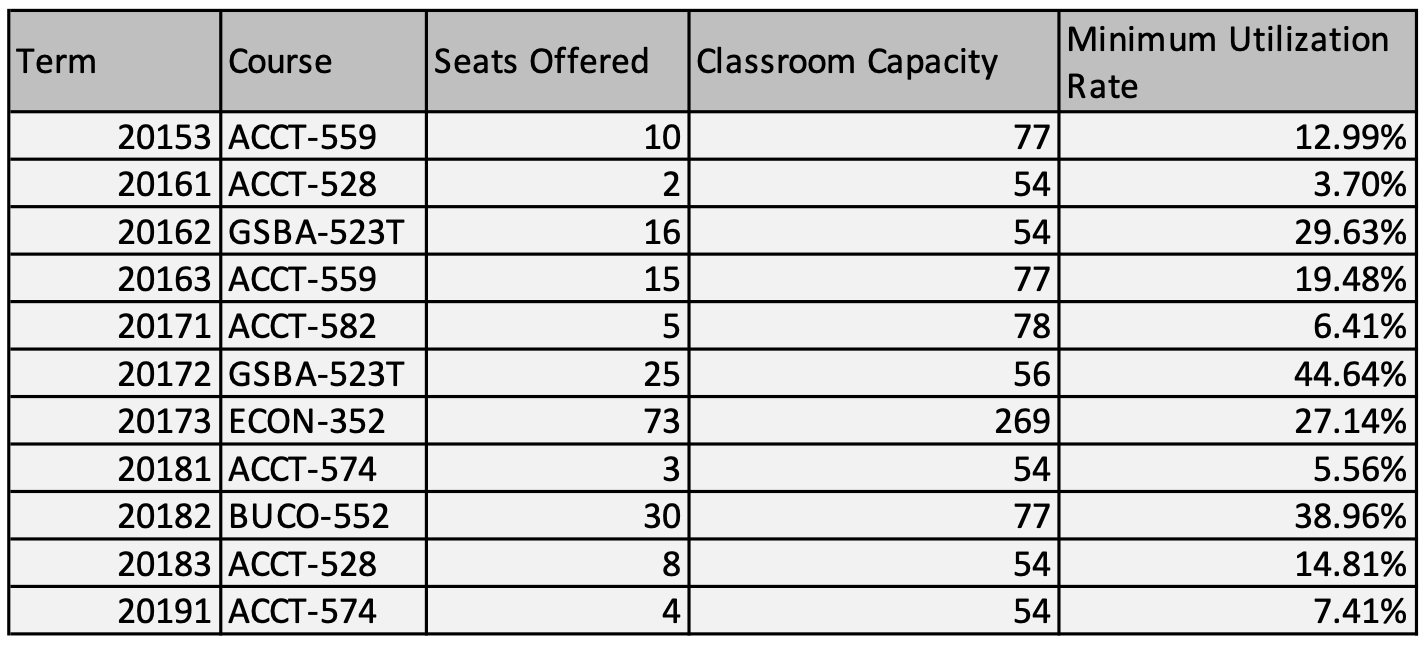


Table 2.1 : Class with minimum utilization rate in each term

The same problem can also be found in the other two metrics. For example, as shown in Figure 2.1, the overall utilization rate fluctuates about 10% from term to term. Considering the fact that the seats offered for classes and the capacity of rooms are in general similar across Spring/Fall/Summer semesters, the utilization rate gap between 2017 Summer (84.4%) and 2019 Summer (77.2%) indicates that there’s room for improvement in terms of classroom usage.

It is also noticeable that the low points of the line graph are all in the summer semesters (2016\_2, 2017\_2, 2018\_2, 2019\_2). This pattern is counter-intuitive, as summer semesters usually have less courses, which potentially leads to more flexible assignment of the classrooms. After investigation, the team discovered that for the purpose of utility saving, Marshall only opens one building, JFF, for summer sessions. As the choice of capacity in JFF is limited, the space of JFF rooms is in general large, and the class size of summer sessions is commonly small, the average utilization rate was pushed down during summer sessions.

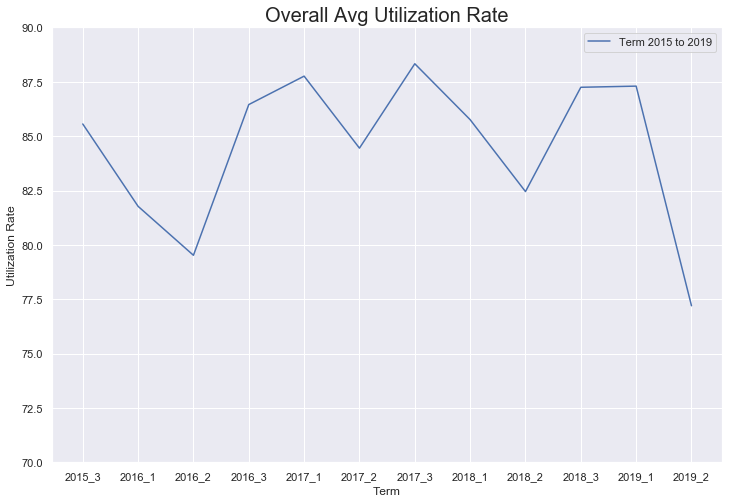


Figure 2.1 : Overall Average Utilization Rate

Finally, looking at the percentage of classes that *inefficiently* utilized the classroom assigned (Table 2.2), the team discovered that in general more than 20% of classes in each term has at least a quarter of space left wasted, which in another way, confirms the team’s assertion that the classroom utilization is one of the critical opportunity for improvement.

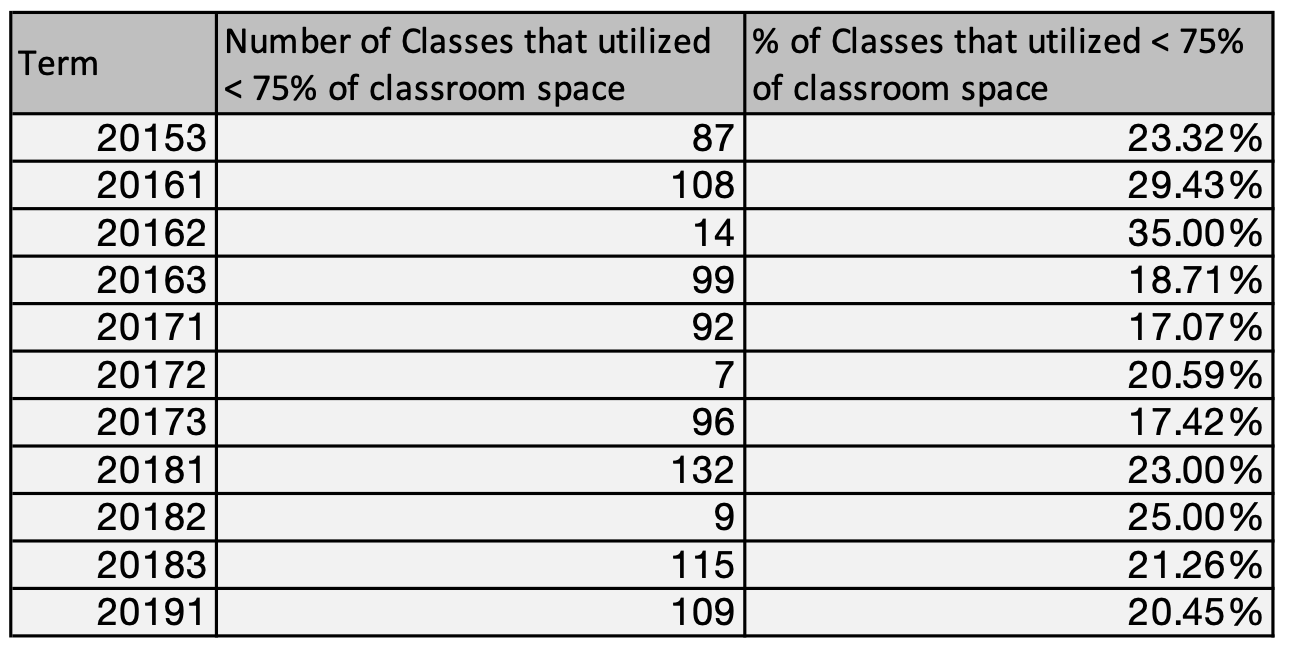


Table 2.2 : Number of classes with utilization rate < 75% in each term

## 2.3 Professor’s Preference Not Satisfied

Faculties are the backbone of any educational system. Given that faculties may be a visiting faculty or might be involved in research activities, it is important to consider their preferences while scheduling class, otherwise it might lead to a hampered teaching experience for professors and even restricted learning experience for the students.

Based on the faculty survey data that was provided, the team narrowed down a few cases to understand the general pattern of faculty preference. Although it cannot be concluded that the survey result is representative of all the faculties, it still acts as a directional guidance to understand the overall trend. The patterns found are discussed in 2.3.1 and 2.3.2.

2.3.1 Avoid Teaching More Than Two Days a Week

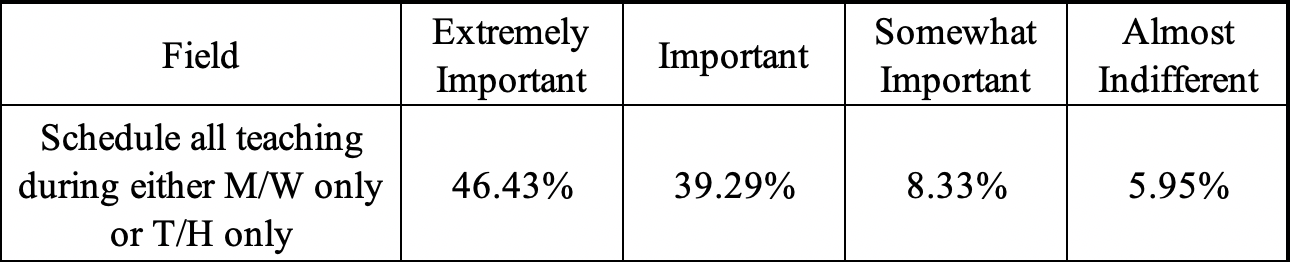


Table 2.3 : Faculty Survey Results

The above survey (Figure 2.3) is on a Likert scale of 1-4 (1 being most important). It is observed that 94% of respondents believe that *scheduling all teaching during either M/W only or T/H only* is at least somewhat important (rated between 1 and 3) and 46.43% of faculty who took the survey believe it is the most important. Also, in the comment page, several respondents left messages stating *maximum 2 days a week is strongly preferred*. It is clear that in general, professors want to avoid teaching more than 2 days a week. This pattern makes sense from a real-life perspective, as it will allow faculty to greatly reduce the commutation time, make a more effective routine and remain productive for the entire week.

To evaluate how past schedules perform in satisfying this particular preference, the team used *the number of professors who have to teach more than two days a week* as a measurement.

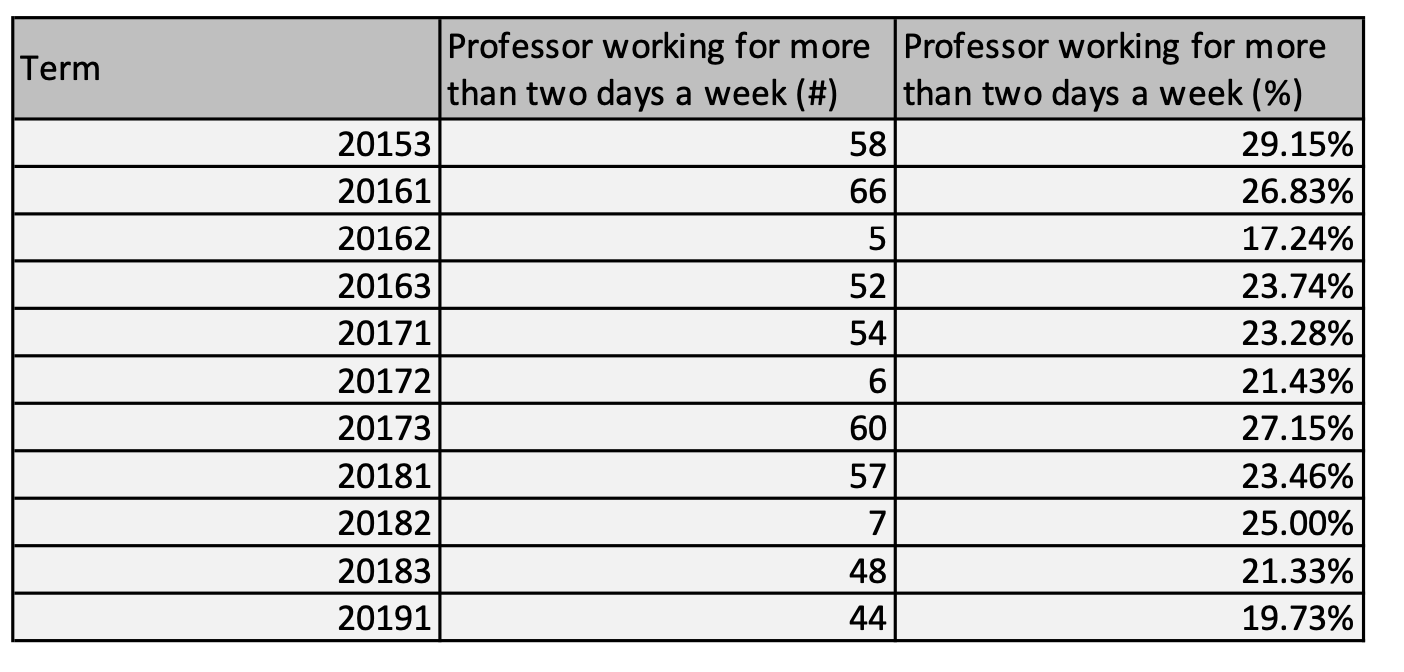


Table 2.4 : Number of professor who have to work more than two days a week

As shown in the table above (Table 2.4), the team can conclude that in general 20% - 30% of professors have to work more than 2 days a week in the past semesters. This number is way above the percentage of professors who are almost indifferent to teaching more than 2 days a week (5.95%). In other words, many professors’ requirements for the number of working days per week are not satisfied by the past schedules.

2.3.2 Desire Back-to-back classes

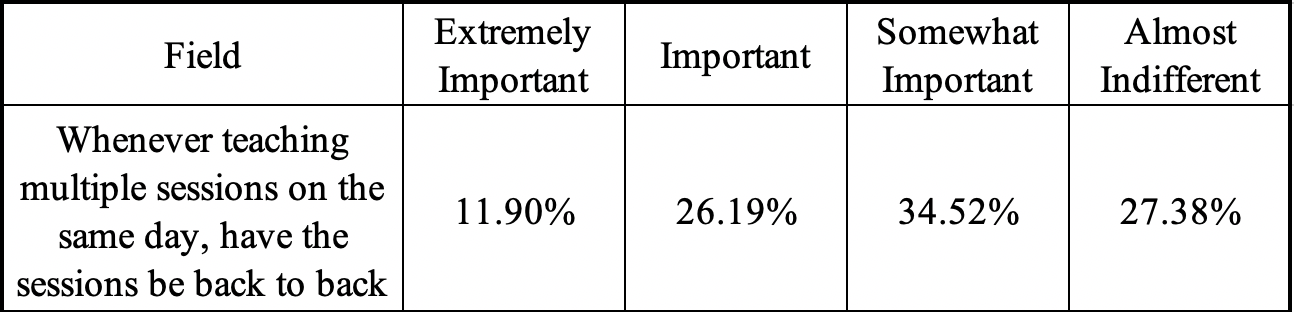


Table 2.5 : Faculty Survey Results

The above survey (Table 2.5) shows that although *teaching back-to-back class* is not exactly the first choice for most professors, 73% of professors believe it is at least somewhat important (rated between 1 and 3). The team also conducted focus group interviews with two professors and, based on the results, it is concluded that many professors prefer back-to-back classes. This pattern is understandable in a real-life scenario, as it minimizes the time professors have to stay on campus, helps them teach better if the consecutive classes are different sections of the same course and makes office hour arrangement more flexible. Although there are cases where some professors prefer long breaks between classes to clear students’ doubts or have a rest, as only general trend is discussed here, these particular cases are not considered for now.

However, as shown by the blue lines in Figure 2.2, the percentage of professors who had back-to-back classes (defined as class with interval less than 30 minutes) are generally around 50% in schedule 2015-2019, while as mentioned previously, around 73% (the red dashed line) of professors believe that having back to back classes is at least in a sense important. The gap between the professors willingness and the actual assignment indicates a potential opportunity for improvement.

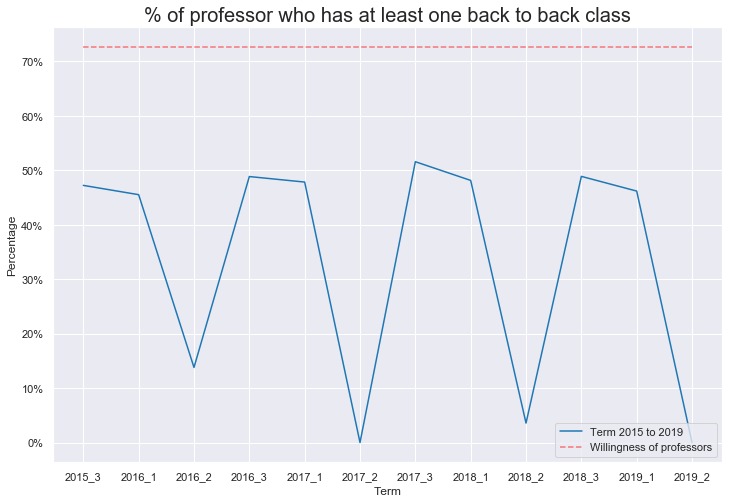


Figure 2.2 : % of professors who have at least one back-to-back class

Additionally, even though many professors do prefer back to back sessions, they still don’t want these sessions to take place in different classrooms. It is really difficult for professors to wrap up, take a rest and go to a different classroom, or in some extreme cases, to a different building, in a short 10-minute break. As expected, several survey respondents left messages stating *having back-to-back classes in the same classroom is strongly preferred*, as it allows professors to take a well-deserved break between their sessions.

However, as depicted in Figure 2.3, the past scheduling of the classroom fails to fully justify this preference, as in each term, around 10% - 20% of professors face the problem of rushing to different rooms during the short break between consecutive class time. (the huge drop in the graph is due to fewer classes in summer term)

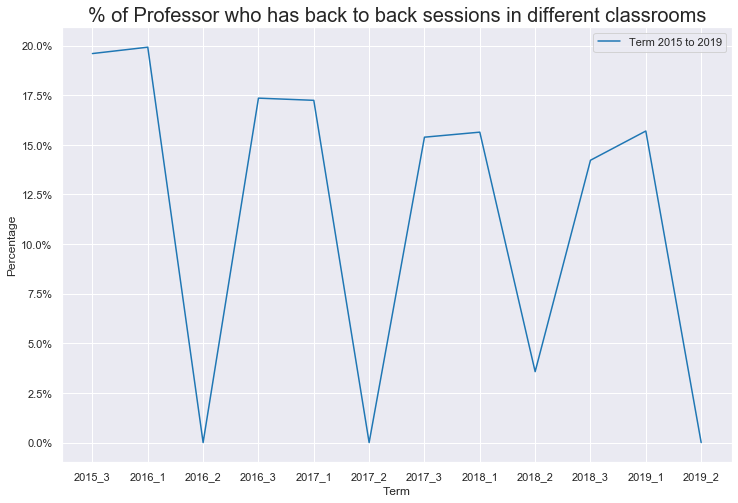


Figure 2.3 : % of Professors who have back-to-back classes in different classrooms

# 3 Optimization Methodology

## 3.1 Input Data

The input data feeded into the optimization tool is composed of 6 parts:

1. A set of available **time slots** to allocate *undergraduate* classes.
2. A set of available **time slots** to allocate *graduate* classes.
3. A set of available **classrooms** to allocate *undergraduate* classes and the corresponding capacity information.
4. A set of available **classrooms** to allocate *graduate* classes and the corresponding capacity information.
5. A set of **classes** (including *undergraduate and graduate*) to be allocated to and the corresponding *level* (undergraduate or graduate), *section index, class index, units*, *number of seats offered*, *session* (full semester, first half or second half) and *instructors* (names of the first and second instructors) information.
6. Three **weights** that end users intend to assign on three performance metrics, including average classroom utilization rate, the number of professors who are allocated at least one back-to-back class and the number of professors who have to work more than two days a week. The weight is between 0 and 1, while 1 means the most important and 0 represents totally indifferent. By default, the three weights are 1, 1, 0.2 respectively.

The sets of time slots for undergraduate and graduate classes can be overlapping, but they are independent of each other, usually with different start time, end time and duration. The set of classrooms for undergraduate and graduate classes are completely different, that is, there are no classrooms that both undergraduate and graduate classes can be allocated to.

The within-level time slots, i.e. timeslots in either undergraduate or graduate level, can also conflict with each other. They take the format of *Day of Week + Start Time + End Time + Session,* eg. MW 11:00 - 12:30 PM (Full Semester). Thus, timeslots such as MW 11:00 - 12:30 PM (Full Semester) and M 8:00 - 11:00 AM (First Half) overlap and conflict with each other.

## 3.2 Output

The output of the optimization tool is composed of 2 parts:

1. A **summary** of the performance of output schedule. The evaluation metrics include:
   * The average classroom utilization rate (defined as U for easy reference)
   * The number of professors who are allocated at least one back-to-back class (R)
   * The number of professors who have to work more than two days a week (Q)
   * The scheduling score calculated by (U - Q + 0.2\*R).
2. A **detailed schedule** for every undergraduate and graduate classes, including the *timeslot* and *classroom* each class is allocated to, together with basic information of the class (*section index*, *class index*, *session, units, seats offered, instructors’ name*) and the *classroom utilization rate*.

## 3.3 Decision Variables, Objective and Constraints

The **decision variable** for the optimization tool is binary, indicating whether a specific class is allocated to a certain classroom at a certain time slot.

The **objective** is the scheduling score mentioned in 3.2 Output, which is calculated by the average classroom utilization rate minus the number of professors who have to work more than two days a week, plus 0.2 times the number of professors who are allocated at least one back-to-back class.

Six **constraints** are included in the model:

1. Hard: Each class eventually only occupies one classroom and one timeslot.
   * More specifically, if it is a double-unit class (i.e. 3 units for graduate and 4 units for undergraduate classes), it takes one of the timeslots categorized as *full semester for 1.5/2 hours twice a week* or *full semester for 3/4 hours once a week*.
   * If the class is a single-unit class (i.e. 1.5 units for graduate and 2 units for undergraduate classes) taken in full semester, then it takes one of the timeslots categorized as *full semester for 1.5/2 hours once a week*.
   * If the class is a single-unit class taken in the first half of semester, then it takes one of the timeslots categorized as *first half for 1.5/2 hours twice a week* or *first half for 3/4 hours once a week*.
   * If the class is a single-unit class taken in the second half of semester, then it takes one of the timeslots categorized as *second half for 1.5/2 hours twice a week* or *second half for 3/4 hours once a week*.
2. Hard: Each classroom can not be occupied by more than one class at the same time.
3. Hard: Each class can not be allocated to any classrooms that fail to accommodate the number of seats offered by the class.
4. Hard: Each professor can not be allocated to more than one classroom or class at the same time.
5. Soft: Make the number of professors who are allocated at least one back-to-back class as large as possible.
6. Soft: Make the number of professors who have to work more than 2 days a week as small as possible.

The precise formulation using mathematical notation can be found in Appendix A1.

## 3.4 Envision of End User Interaction

Shannan Faris, Assistant Dean of Institutional Research and Academic Administration at the USC Marshall School of Business, and her team are the end users of the proposed optimization tool.

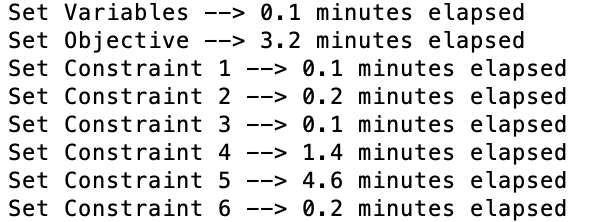
To use the optimization software, Shannan and her team should take the following two steps:

1. Open a Command window (in Windows) or a Terminal (in mac). Navigate in the command line to the directory containing *optimize.py* using *cd directoryName*.
2. Type *python optimize.py input.xslx output.xlsx weight1 weight2 weight3* in the command line. Finally, press *Enter* on the keyboard to run the command. If successfully solved, the output file can be found under the same directory as *optimize.py*.
   * *input* represents the name of the input file and *output* represents the name of the output file.
   * The last three values are the weight assigned to the three evaluation metrics: average classroom utilization rate, the number of professors who are allocated at least one back-to-back class and the number of professors who have to work more than two days a week. The weights can be any decimals between 0 and 1, with 1 indicating the corresponding feature is very important and 0 as not important at all. If the end user didn’t input any weights, then the software will by default interpret weight1 = 1, weight2 = 1 and weight3 = 0.2.

The software prints text to help the end user either track the optimization status or identify the error. Below are three examples:

1. If any of the input/output files does not exist, the software will notify the end user by printing *File “{FileName}” is not found* in the command line.
2. If any of the key input information is missing, such as input file or output file, then the software will notify the end user by printing *Correct syntax: python optimize.py inputFile outputFile weight1(optional) weight2(optional) weight3(optional)*
3. If the optimization successfully started, the software will update the end user with current progress and the time consumption of each part. For instance, after setting up the objective of the model, the tool will notify the end user by printing *Set Objective --> XX minutes elapsed*. When the optimization work is finished, the software will inform the end user by printing *Results in "{outputFile}"*. (As shown in Figure 3.1)

## 



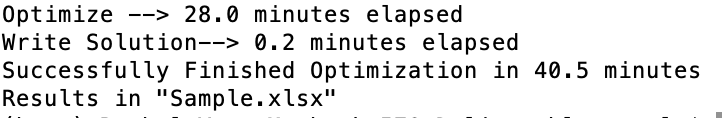


Figure 3.1 : Sample of end user interaction

4 Optimization Results

In order to testrun the model and estimate the potential gain from the tool, the team ran the optimization model using sample data of classes in the 2019 Spring term, with 533 classes, 38 classrooms and 241 timeslots. A new suggested schedule was produced and evaluation analysis had been performed on this particular schedule. Four performance metrics are used to help evaluate the optimization model and allocation strategy as encompassed in the optimization algorithm. The metrics will be visualized using multiple graphs in 4.1, with an extra yellow line indicating the new value, a blue line indicating the old value and an arrow indicating the difference. The final value of the four metrics are as following:

1. Scheduling score (weight1\*U - weight2\*Q + weight3\*R, with all weights set as default) = 118.17
2. The average Utilization Rate (U) = 89.57%
3. The number of professors who are allocated at least one back-to-back class (R) = 143 (64.13% of total professors)
4. The number of professors who have to work more than two days a week (Q) = 0

The potential gain from applying optimization methodology is explained metric by metric in 4.1

4.1 Potential Gain

4.1.1 Efficiency in course scheduling

An essential benefit from using the model is the huge increase in the efficiency in course scheduling work. Originally the scheduling work of allocating courses and classrooms has to begin almost one year before the semester begins, done by administration staff doing it manually. Now with the existence of the schedule recommending model, the work will be done in a timely manner. For as little as 40 minutes the model is able to produce a free-from-conflict schedule for all 534 classes in 2019 Spring Semester, while pushing the classroom utilization rate to the highest, increasing the number of professors who have back-to-back class and decreasing the number of professors having to work for more than two days in a week as much as possible. In a word, the tool would, to a large extent,increase the scheduling efficiency of the administration office.

4.1.2 Efficient Space Utilization

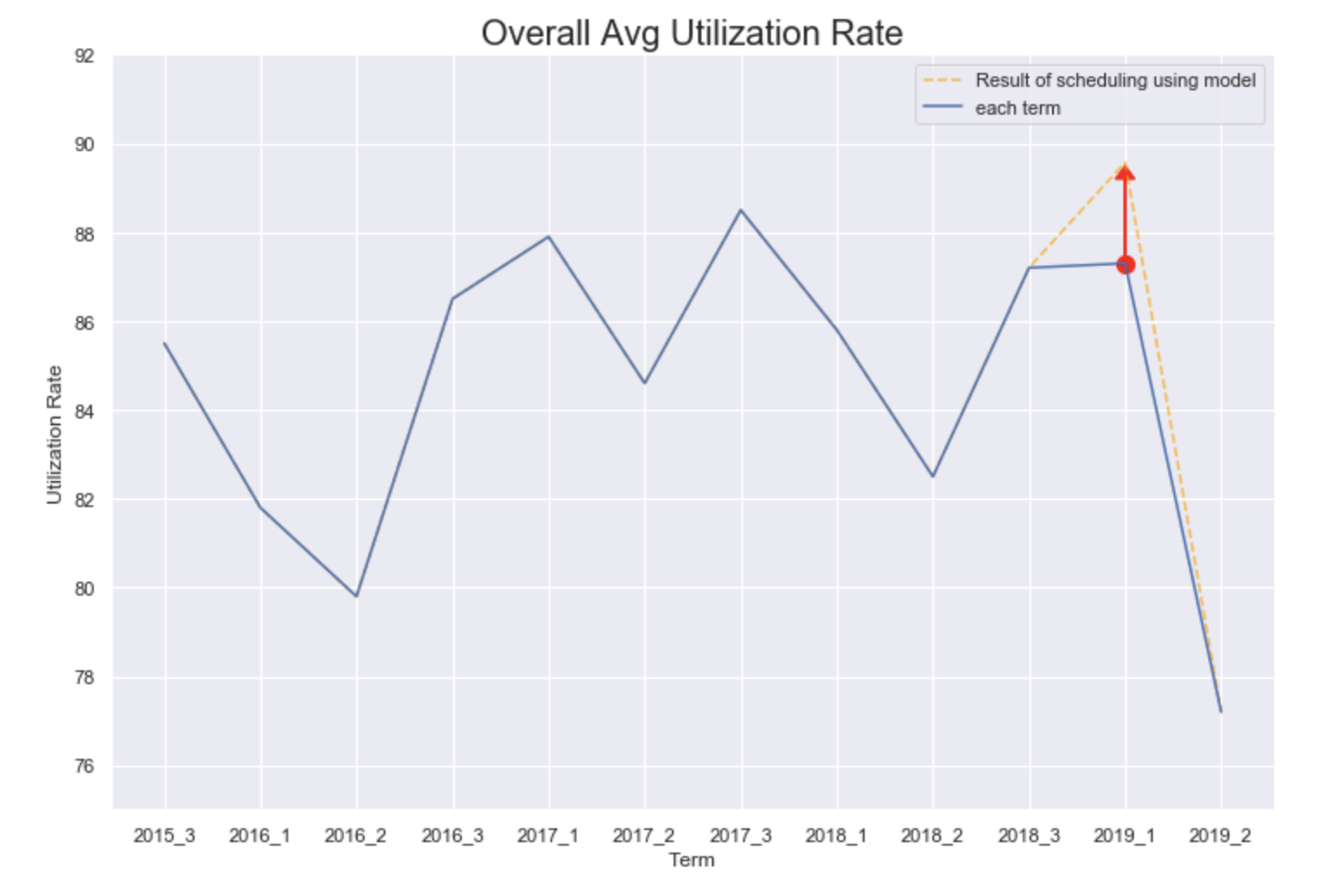


Figure 4.1 : Overall average utilization rate

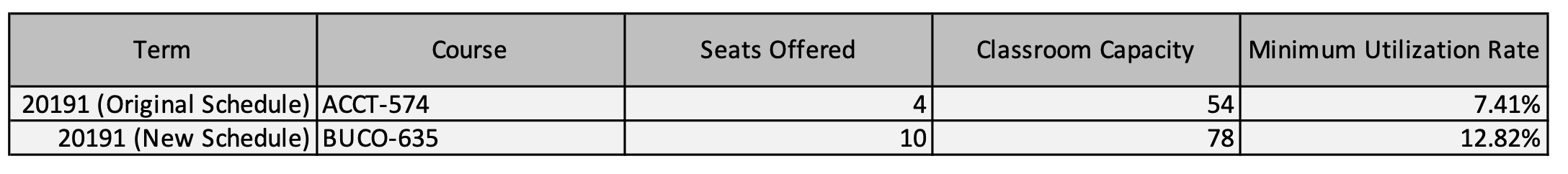


Table 4.1 : Class with minimum utilization rate

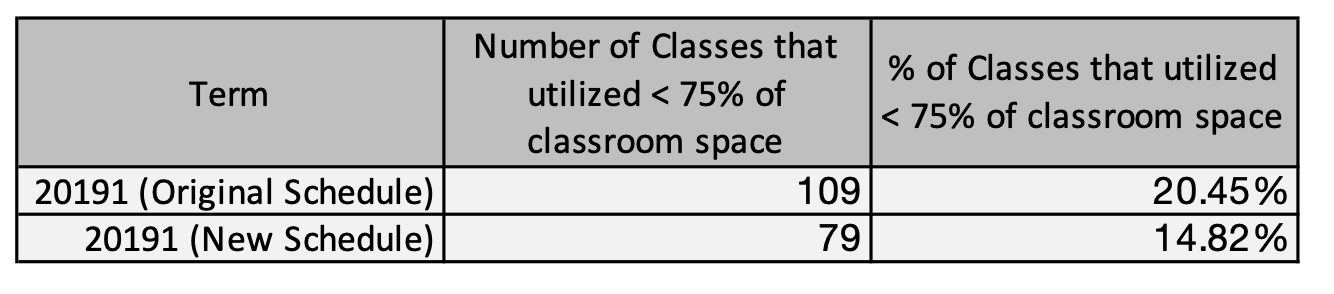


Table 4.2 : Number of classes with utilization rate < 75% in each term

As for the first objective, which is to maximize the utilization rate of each classroom, the model performance is shown in figure 4.1. The blue line indicates the original average utilization rate before using the model, while the yellow line indicates the new average utilization rate from the model output. There’s an obvious increase in the average utilization rate using the model, as depicted by the red arrow. The original schedule has an average utilization rate of 87.3% in the 2019 Spring Semester, and after using the model to conduct class scheduling, the average utilization rate increased to 89.57%. In addition, as shown in table 4.1 and 4.2, the original schedule has the minimum utilization rate of 7.41% and is having 20.45% of the classes with utilization rate < 75% in the 2019 Spring Semester, and after the model was applied, the minimum utilization rate increased to 12.82% and the number of classes with utilization rate < 75% has dropped to 14.82%.These metrics indicate that the model is indeed helpful in terms of improving space utilization. The higher the average utilization rate is, the more efficient are the spaces in Marshall classrooms being used in general.

4.1.3 Number of professors who are allocated at least one back-to-back class

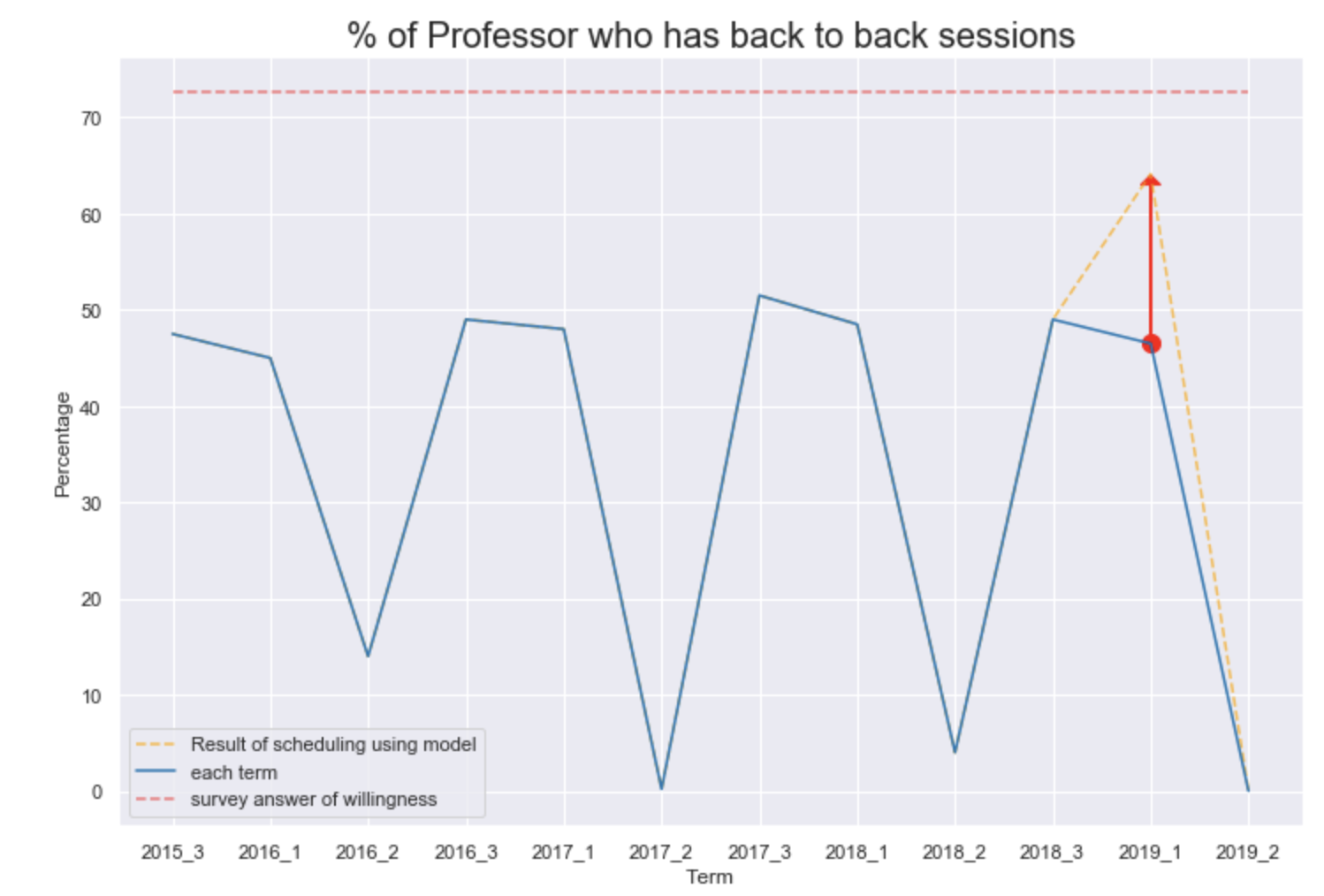


Figure 4.2 : % of Professor who are allocated at least one back-to-back class

As for the second objective, which is to maximize the number of professors who are allocated at least one back-to-back class, the model performance is shown in figure 4.2. The blue line indicates the original number of professors who have at least one back-to-back courses, while the yellow line shows the new number outputted from the model. It’s clear that there’s an obvious increase: Only 46.5% of the professors were assigned at least one back-to-back course in the original 2019 Spring schedule, while the number increased to 64% after applying the optimization tool. It confirms that the model is effective in terms of successfully increasing the occurrences of back-to-back courses in the schedule.

4.1.4 Number of professors who has to work more than two days a week

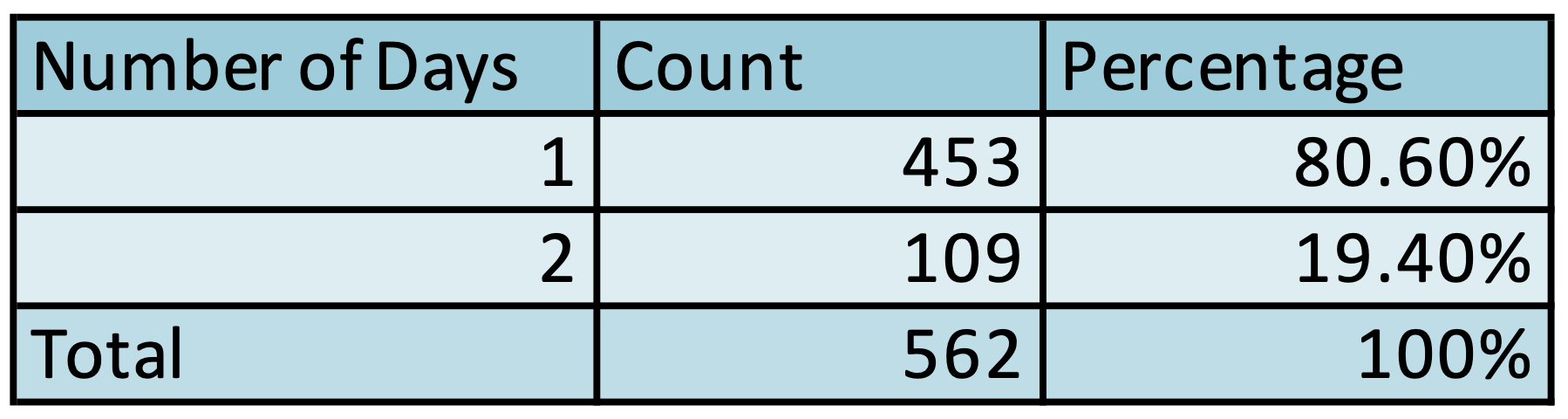


Table 4.3: % of Professor who has to work more than two days a week

As for the third objective, which is to minimize the number of professors who have to work more than two days a week, the model performance is shown in table 4.1. Compared to the original 2019 Spring schedule, which there are 44 professors who need to work more than two days a week, in the new schedule 80.6% of the professors work for only one day a week, and the remaining 19.4% professors work for two days a week. More importantly, the output schedule makes it possible for all professors to work at most two days a week, a feature strongly desired by most faculty. The analysis above indicates that the model perfectly done its job to satisfy this particular feature.

5 Discussion

5.1 Appropriateness of Methodology

The main methodology used in the model is to base the model on realistic situations, making the model useful in helping with the course allocation process for school scheduling administratives, and reusable in any semester.

5.1.1 Rational Behind Choice of Input Data, Output Data, Objectives and Constraints

The input data contain basic information of courses, classroom and related professor. These data are essential when conducting course scheduling, they are needed in order to correctly assign each course taught by a certain professor to a certain classroom and time slot without overlapping with other courses taught by the same professor at the same time. For the purpose of creating as fewer decision variables as possible and saving the computation power, the input data mentioned above should be separated by Undergrad/Graduate level courses and provided in two different worksheets in a single MS Excel file, under specific columns and rows as shown in the sample input Excel file attached with this report. All of these are already in the school scheduling office’s possession, so they are easy to collect without having to spend additional fetching costs or laboring effort.

The output of the model is shown in clear tables in separate worksheets for Undergrad/Graduate schedules, with all the key information needed for the administration office for scheduling: the *section index*, *course name*, *units, classroom* assigned, *time* assigned and *instructor* information. Additional information such as *seats offered, classroom capacity* and *utilization rate* are also offered so that the end user can easily double-check capacity conflicts and get an rough idea of classroom utilization. The separated worksheets are easy to interpret, ready to use and convenient to extract key information such as session index. The optimization metrics in a separated worksheet also indicate the key performance of the model, making it easy for users to understand the overall utilization rate as well as other desired features of the scheduling in a quantitative way.

The objectives of the model are to maximize the average utilization rate, minimize the number of professors who have to work more than two days a week, and to maximize the number of professors who have at least one back-to-back class at the same time. Additionally, the model allows flexible weighting of the three metrics, as it intakes the corresponding weights as input. That being said, the end users can easily change the weight assigned on any metric by simply typing the weight wanted in the command line, which is free from changing the model and can be accepted by non-technical users with little knowledge of Python. The reason behind the choice of objectives is because the main goal of the model is to offer a schedule focusing on optimizing the overall efficiency of classroom utilization*,* while taking two specific professor preferences, the desire for working at most two a week and for back-to-back classes into account.

The constraints of the model are listed in Section 3.3 in detail. The hard constraints are basic restrictions needed to create a conflict-free schedule, while soft constraints are developed to help satisfy the professors’ preference as much as possible.

5.1.2 Key Assumptions

While conducting data preparation for model building, the sample data was built on below assumptions:

1. Before class scheduling starts, information including the instructor, the seats offered as well as in which session each class will take place is already decided.All the graduate level courses either take 1.5 or 3 hours, while all the undergraduate level courses are either take 2 or 4 hours
2. Each classroom either only accommodates graduate classes or undergraduate classes. Courses in 1.5 units will not be allocated in evening time
3. Marshall classrooms are closed on Friday evening and no courses can be assigned after 6:30 PM on Friday. All courses under Marshall school of business will only be assigned to classrooms in Marshall business school owned buildings

All the assumptions listed above were set with regards to simplify the scheduling process, improve student/professor satisfaction or to comply with current Marshall convention.

5.1.3 Model Weaknesses

While several constraints are taken into account when building the model, there are still several remaining weaknesses yet to be solved in the future:

1. **Non-personalized preference**: Each professor’s preference and availability are not taken into consideration. The model is simply following the generic preference patterns obtained from the anonymous survey, while ignoring the availability restriction or personalized preference of each professor. In addition, the differences in teaching requirements for full-time and visiting faculties are also not considered in this analysis. Being assigned to inappropriate or unavailable time slots for some professors may not go well with their schedule or teaching style and as a result can hamper a good learning experience.
2. **Neglecting course popularity and Bundled Courses**: A few popular classes and mandatory courses must be taken together should not be assigned to the same day and time. However, this has not been taken into consideration by the model. It may result in students dropping out from a popular or must-take course due to scheduling conflicts, which may create a negative sentiment and prevent students from having the best education needed.
3. **Pre-decided classrooms**: All classrooms are pre-decided to either only take undergraduate or graduate level classes to improve the computational efficiency, but at the cost, the results might not be able to capture the best combination received by considering all the combinations.
4. **Input timeslots need manual adjustment**: All time slots are listed in the input file rather than being auto-generated by codes. Also, they are developed based on the standard duration of courses. That being said, if the time slots are to be modified (for example, the standard length of Undergraduate/Graduate level courses is changed in the future), then the software requires the user to adjust the time slots manually in the input file, which can be in a sense troublesome.
5. **Exceptions need manual adjustments**: As mentioned in point 4, the model only considers the standard length of classes. However, there are very rare cases where undergraduate classes are 6 hours or 3 hours. Exceptions like this can not be automatically justified by the model and requires the end user to manually adjust based on the output file.

ii) Final Recommendation

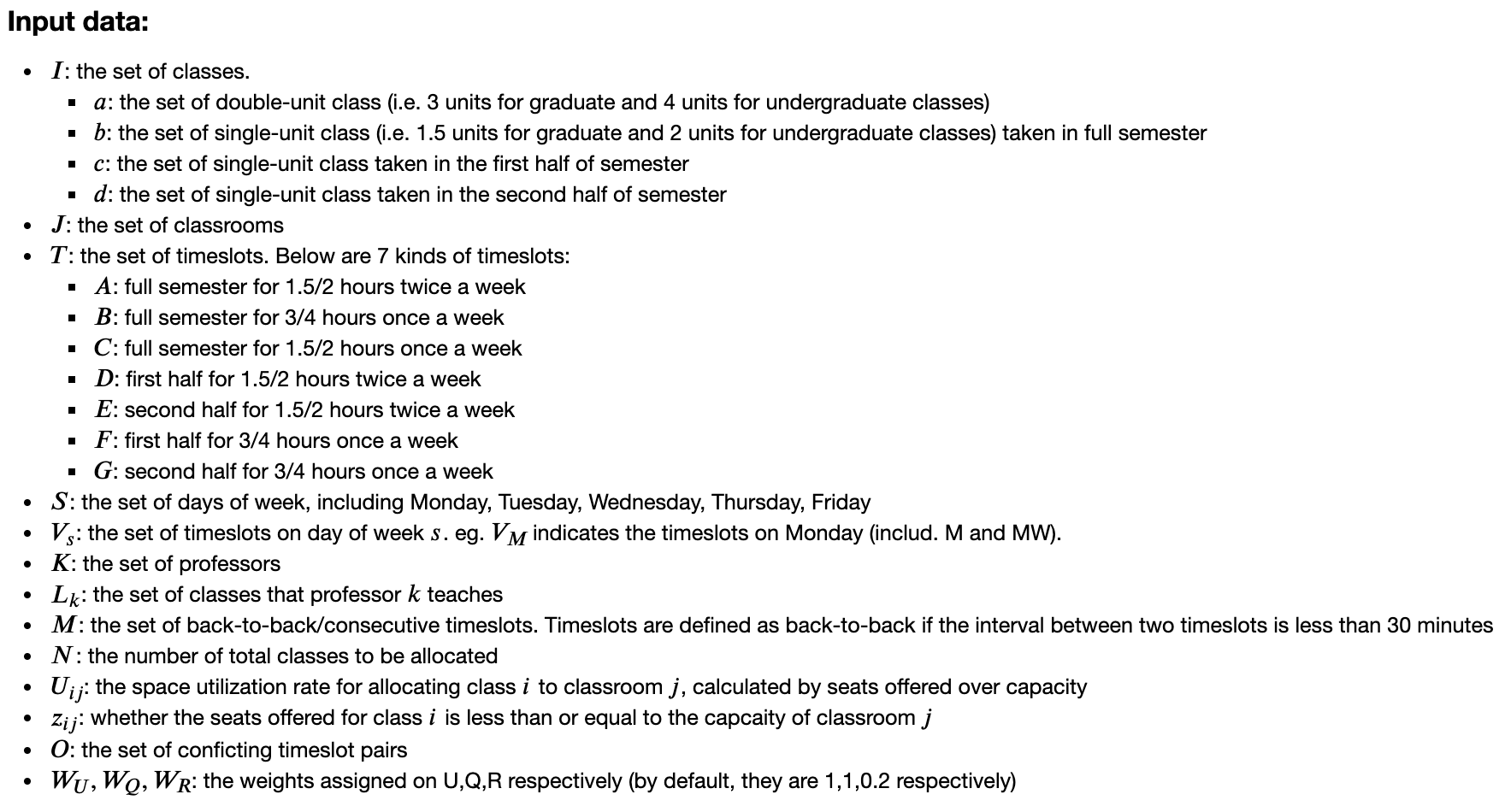
The current approach of class scheduling consists of utilizing historical course allocations and manually dealing with changes as well as individual department preferences. As the process is time consuming and inefficient, the proposed model could not only release the administration office from the headache of the complicated course scheduling, but also suggest the schedule with best utilization and satisfy the professors.

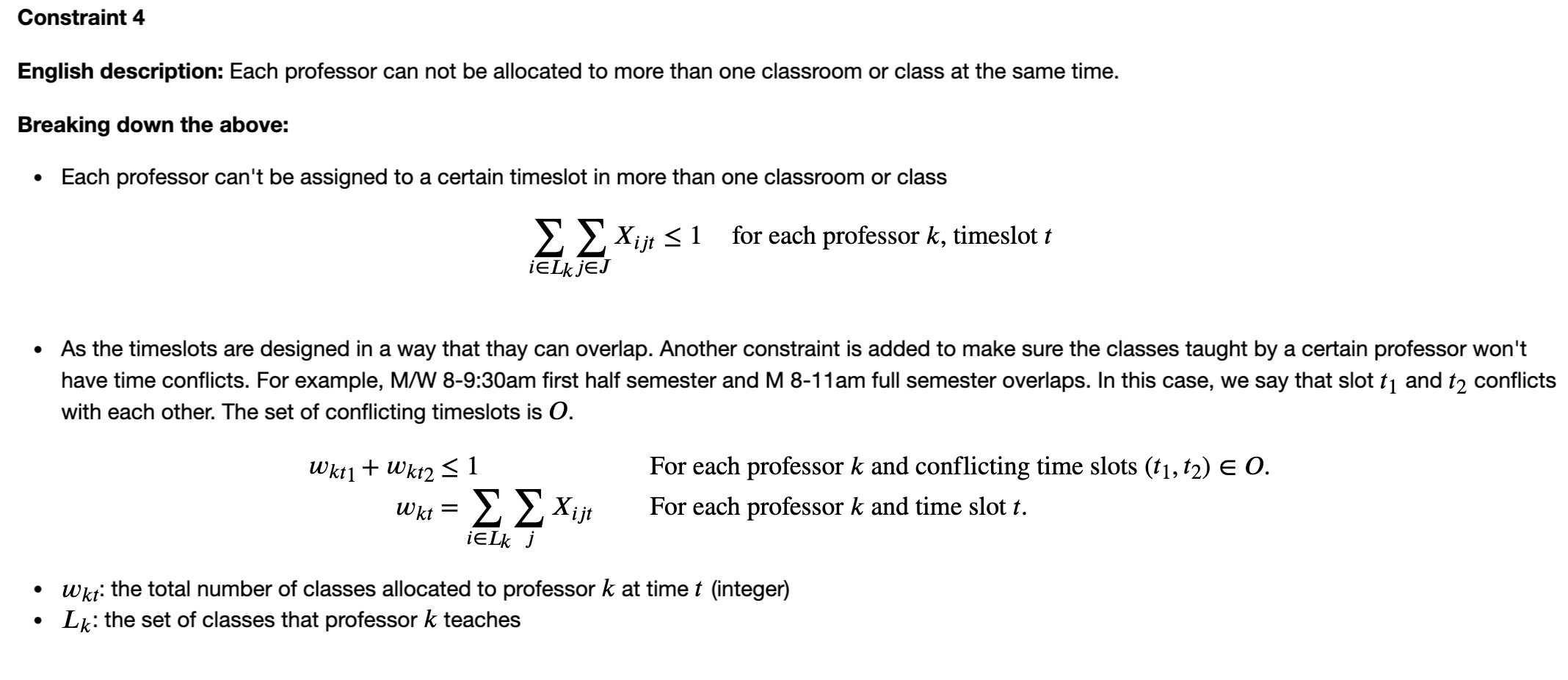
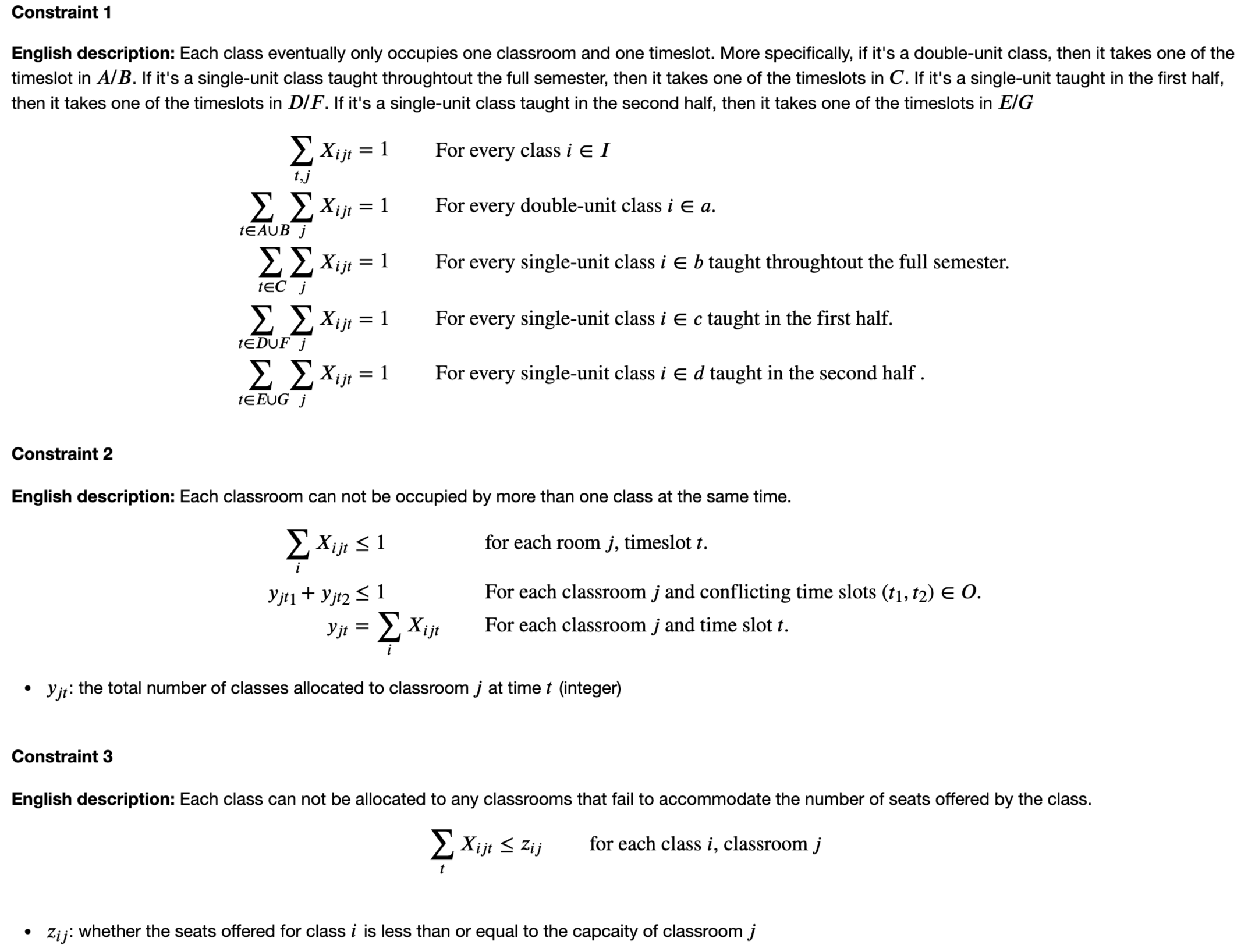
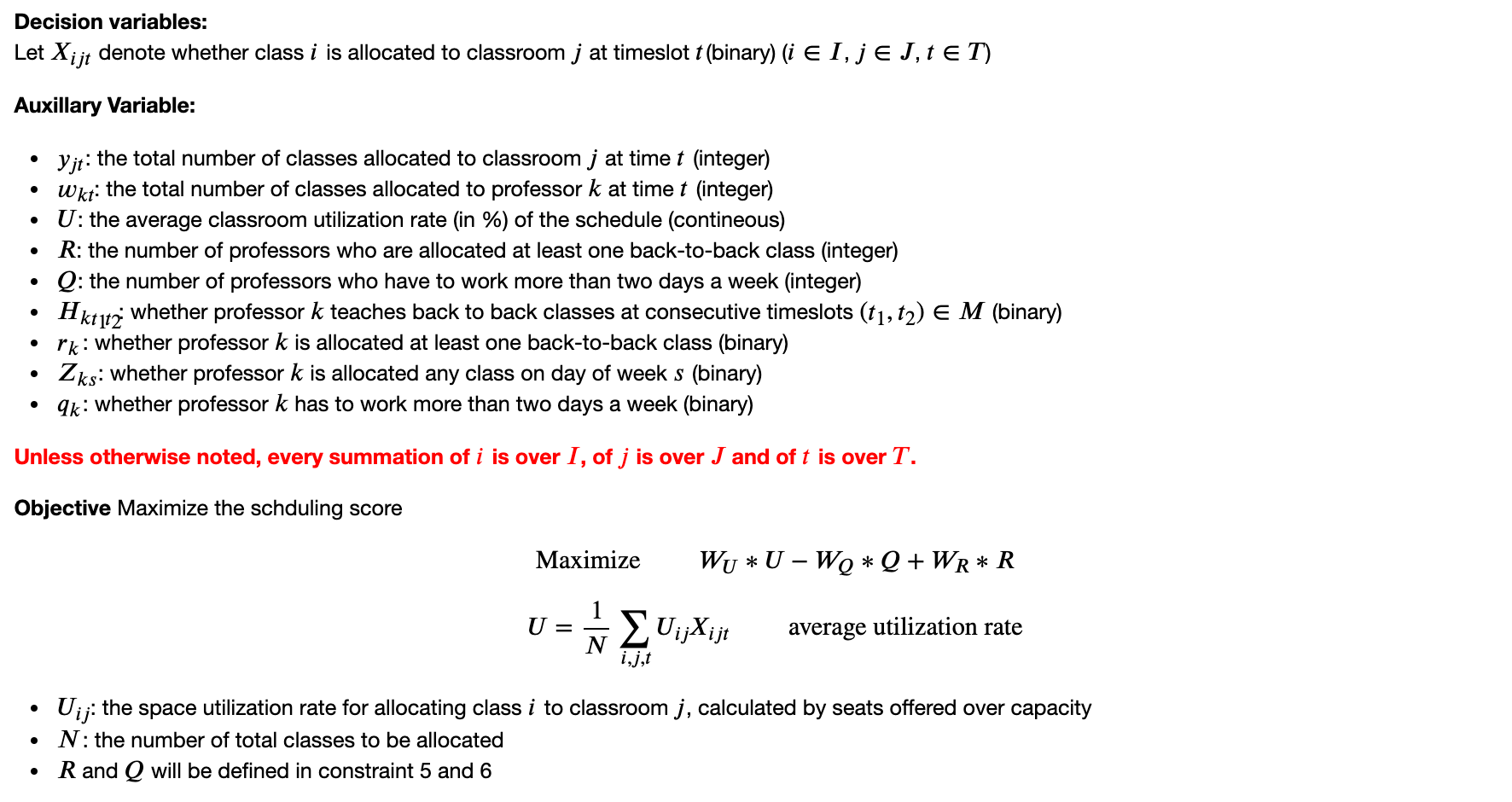
While it’s suggested for Shannon and her team to utilize the model, there’s a series of final recommendations that is to be given:

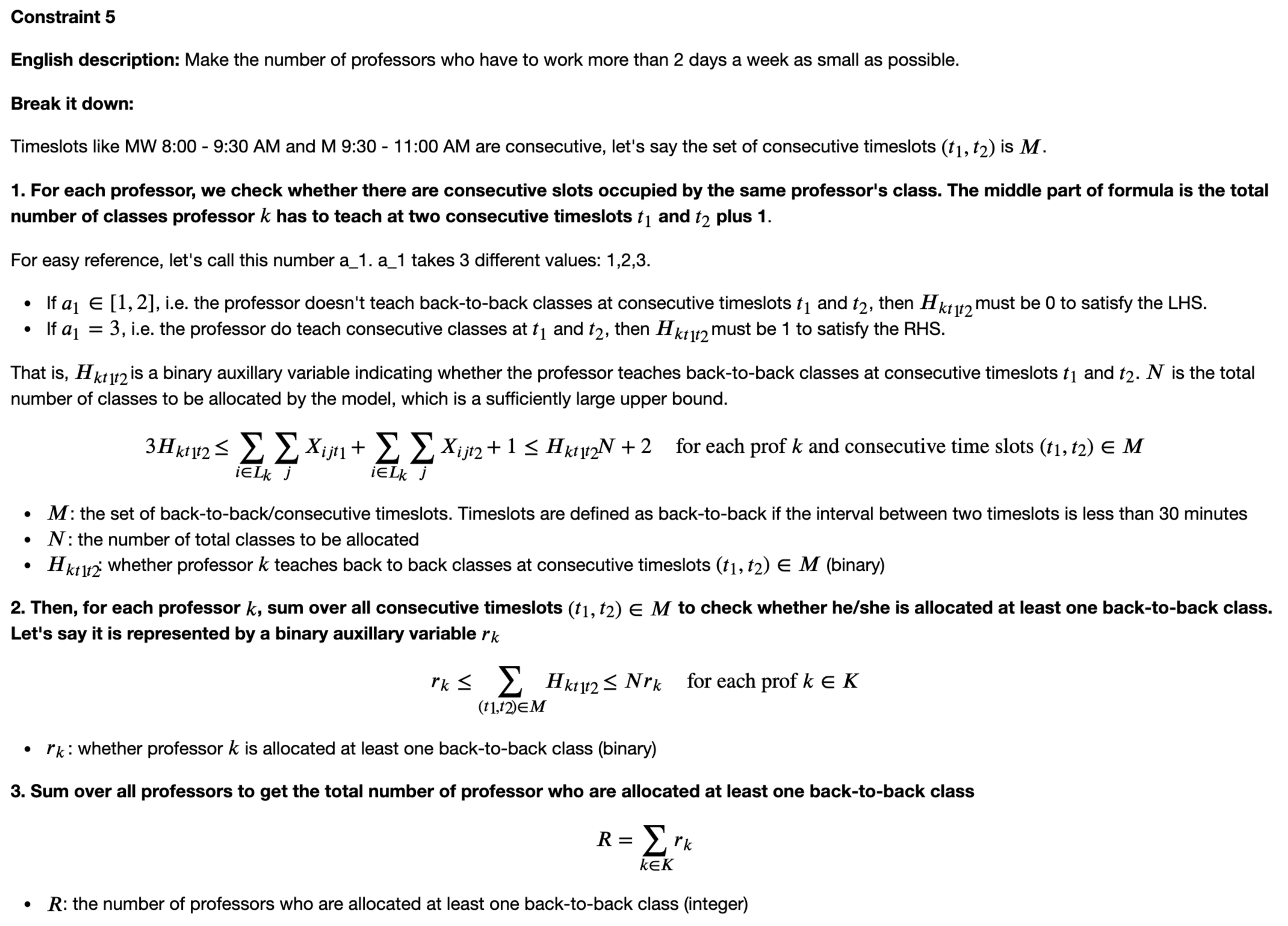
* **Start using the optimization tool with piloting a small section of classes**. Even though the detailed description on how to use the model and the illustration of the sample output and result metrics generated by Spring 2019 data are listed above, it’s still hard for the Marshall administration office to trust the model and utilize the tool to the fullest immediately. It is recommended to start using the model with a smaller size of input data, and familiarize the team with the tool and the way to interpret the output data. For instance, set the input data for the model with only MBA courses when first trying out the model, and check to confirm if the schedule is proper, free-of-conflict and achieves the preference-related goals. In this case, the model could be gradually used to the fullest.
* **Set the weight based on scheduling needs or school policy**. Such as space utilization, back-to-back classes and number of days to work can be considered with different importance by easily inputting the corresponding weight in the command line. The logics and restrictions related to these features might change from term to term based on school policy or department needs, so it would be helpful for Shannon to tune the weights by conducting sensitive analysis . That is, Shannon and her team can gradually change the weights allocated to each feature and see how the performance metrics and class schedules change accordingly, and finally select the set of weights that gives the most desired schedule performance. By default, the weight for the average classroom utilization rate is set to be 1, the number of professors who are allocated at least one back-to-back class is set to be 1, and the number of professors who have to work more than two days a week is set to be 0.2.
* **Adopt the suggested schedule and manually address additional requirements**. There will always be course changes and additional scheduling requirements in every term, such as professor availability from individual departments, and faculty preferences on time slots or classrooms, which should be taken into account when conducting course scheduling but not captured by the model. Thus, in addition to the output schedule generated by the optimization model, special requirements still have to be handled manually by the administration office.
* **Perform data cleaning to increase model efficiency**. It took the proposed model 25 - 40 minutes to run on all the data of the 2019 Spring term, with 533 classes from 7 departments and 22 academic programs, 38 classrooms and 241 timeslots. Since the bigger the data size is, the longer it takes to finish running the model and produce the desired output, it’s suggested to perform necessary data cleaning to decrease the data size. For instance, splitting the input undergrad/graduate courses, classrooms and time slots into 2 - 4 non-overlapping parts will massively decrease the size of each individual dataset, and in that case, it would be possible for the model to run on separated parts parallely in order to increase model efficiency. Finally,Shannan and her team can simply concat the optimization output of all parts to create the full schedule.

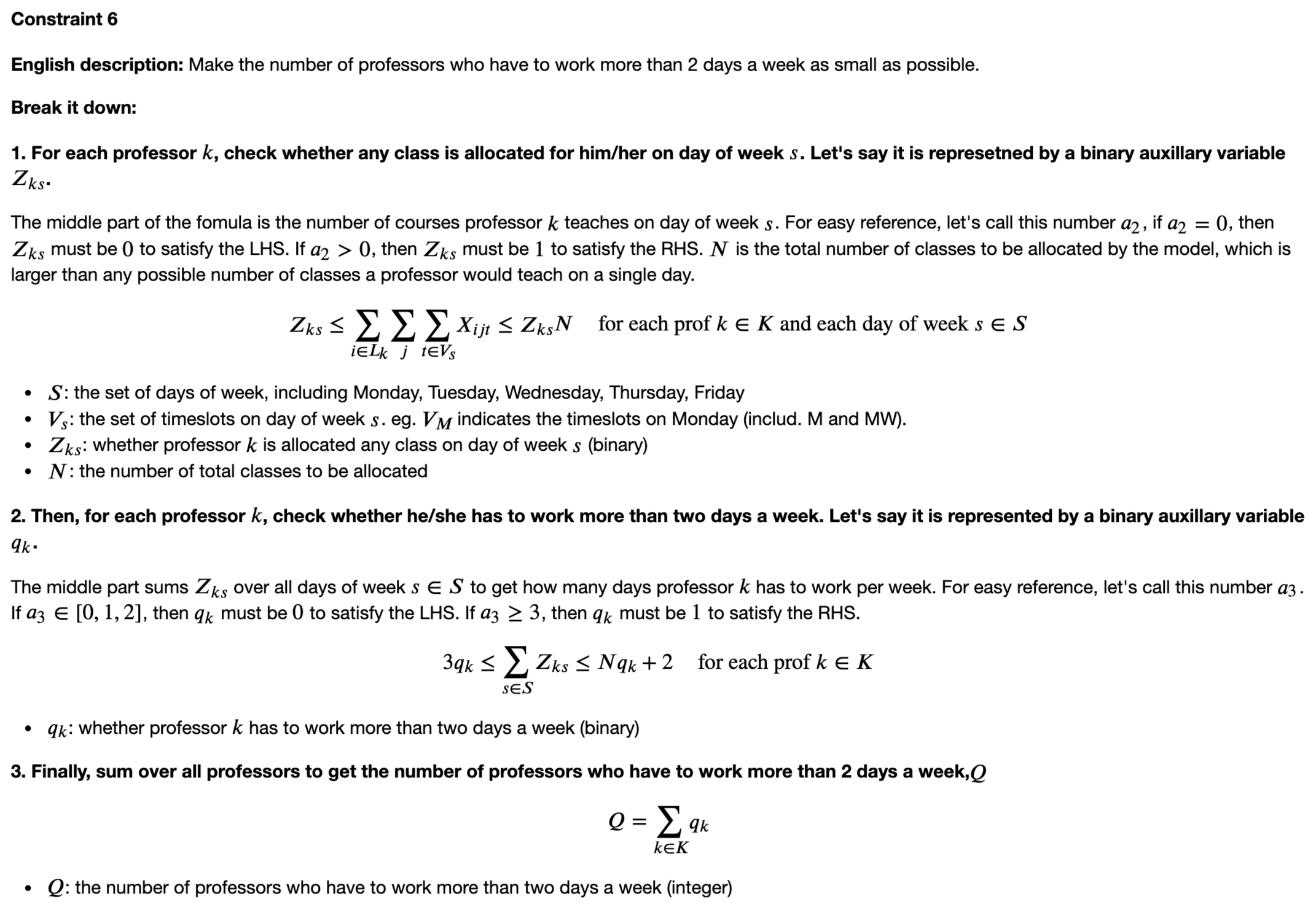
Appendix

## A1. Mathematical Formulation









## A2. Discussion of Technical Details

Reasoning Behind Assumptions:

* The team assumes that all the courses that we feed into the model have an assigned instructor, and the seats offered as well as in which session (full semester/first half/second half) each class will take place is already decided.. This assumption follows Marshall current convention and makes sense in a realistic term. Take professor as an example, if these courses are scheduled in advance, later when the department assigns an instructor to these courses later, there is a good chance that it can violate some hard constraints. For example, when a session without a pre-assigned instructor is allocated to Monday 8.00am~9.30am, and later the department assigns a professor to teach this class. However, there is a chance that this professor has a conflict in schedule and consequently Marshall needs to re-allocate the schedule.
* The team assumes all the undergraduate courses are either 2.0 units or 4.0 units, and all the graduate courses are either 1.5 units or 3.0 units. The team assumes that only double-unit (4 units for undergrad and 3 units for grad) classes will be 4/3 hours a week, and all other unit classes are either 2/3 hours a week for full semester or 4/3 hours a week for half semester. This assumption complies with the pattern with current Marshall classes and there are very few exceptions. Shannon and Hal can schedule these extreme cases manually based on the model output which greatly simplifies the model.
* The team assumes all the undergraduate courses and graduate courses take place in completely different classrooms. Even though it is possible and better to schedule the undergraduate and graduate courses together in all classrooms, it would save us more than 90% of time if we schedule these courses separately. Also, this assumption is in line with the way Marshall scheduled classes in the past few years, which prevents the model from resulting drastic changes that could not be easily accepted by the team. For example, Marshall only allocates graduate courses to JKP, and allocates undergraduate courses to other classrooms. The team assumes that no 1.5 units graduate courses will occupy an evening slot, however 2 units course for undergraduates can still occupy an evening slot, which is, again, complied with the rules and convention of Marshall.
* The team assumes no courses will take place on Friday evening, as in reality, all Marshall classrooms are closed on Friday evenings.
* The team assumes all courses will take place in and only in Marshall classrooms.This assumption is based on Marshall’s preference, as Shanon and Hal mentioned that the team would love to use its own classrooms as much as possible instead of using USC classrooms so as to minimize unnecessary cost.

Discussion of weakness and future work:

* **Non-personalized preference**: So far in our model, the team considered professor’s preferences on an overall aspect, instead of taking every professor’s own special preference and availability into account. The high overall performance doesn’t necessarily mean everyone is satisfied and it’s possible that some professor is assigned to a timeslot that does not work for him/her. This weakness is critical and needed to be handled. To solve this problem, what the team needs to do is to conduct a survey to collect professors’ individual preferences over certain features and their availability over time slots, and then set professor’s availability as a hard constraint and individual level preference score as a soft constraint. If given more time, the team will take preference and availability personalization into consideration.
* **Neglecting Course Popularity**: Another problem the team is now facing is that we didn’t consider the correlation between different courses. Some courses are popular among students or required by school policy and many students are planning to take these courses in the same semester. However, we didn’t add constraints in our model to make sure these courses will not conflict with each other. In future work, we will prepare a dataset that indicates all the courses that should not happen on the same day and the same time, including required courses and popular elective courses, and convert these into hard or soft constraints (depending on how strict we want the constraints to be).
* **Pre-decided classrooms**: We scheduled undergraduate courses and graduate courses separately, which increased our efficiency tremendously at the cost of a maybe lower, but acceptable performance. We can increase the performance by scheduling these undergraduate courses and graduate courses together, but that will also result in a huge increase in the model running time. Therefore, whether we update our model or not is totally according to user’s willingness. If our users think finding the optimal solution outweighs the running time, or they are equipped with advanced computers that can speed up the optimization time, then the team can update the model to make sure all classrooms are considered for each class.
* **Fixed time slots**: In our current model, the time slots for both undergraduate courses and graduate courses are pre-assigned and fixed. However, if in the future, Marshall decides to change the time slots for courses, we can update the input file for them.
* **Neglecting unpopular features**: Our current model has done a reasonable job in improving utilization rate, increasing the percentage of professors who have back to back classes and decreasing the number of professors who have classes for more than two days. However, there are some other features that we neglect at the moment but can be fully addressed in the future, such as, unpopular class time for students. The team can address this particular example by setting the number of professors who have early morning classes or late evening classes as a soft constraint and minimize this number in objective function. As a result, the scheduling score will have to balance the trade-off between number of days for each professor to work and assigning unpopular classes to unpopular times.