Bilateral Weighted Scheduling Optimization with Probabilistic Start Time Estimation

For this project, we have developed a model for scheduling appointments based on estimated appointment durations. This model is able to determine the optimal start time for different appointment types based on the previous appointments for that day, the estimated duration of those appointments, and the no show rate. By increasing the accuracy of scheduled appointments, we hope to increase the utilization and decrease the makespan, or time to the first available appointment. Our model consists of three primary parts: the duration model, the scheduling engine, and the simulator. The duration model estimates the duration of appointments based on descriptive factors such as appointment type and provider, the scheduling engine creates a schedule based on the estimated durations and provider constraints, and the simulator tests how the schedule would have performed when given the realized durations of appointments. This simulation method is flexible and allows not only for testing of the model but can also be modified for what-if analysis—i.e. what if we are able to reduce the variance of appointment lengths by 10 percent.

The Duration Model

Introduction

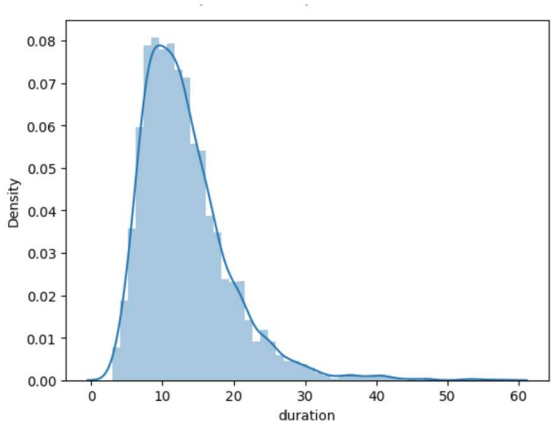
The duration model estimates the probability distribution of the number of appointments based on predictive factors such as the provider and appointment type.

Kernel Density Estimation

It achieves this by using the historical data of appointment durations to fit a kernel density estimator. Kernel density estimation (KDE) infers the continuous distribution of the data based on discrete samples. It can be visualized best with a histogram. Figure 1 shows an example histogram of appointment durations with a line representing the kernel density estimation. For each appointment type, a different probability distribution of appointment durations is fit using KDE.

Figure 1

Example Histogram with KDE



On a more technical level, KDE is a non-parametric way to produce a smooth estimate of the probability density function of a continuous random variable. It achieves this by placing a kernel function (usually Gaussian) at each point and summing them to estimate the distribution. This is expressed mathematically as:

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

where n is the number of points; K is the kernel function (usually Gaussian); h is a hyperparameter known as the bandwidth, which controls the smoothness of the estimate, and x_i represents the individual data points.

The Scheduling Engine

Introduction

The scheduling engine builds a schedule based on the estimates of appointment lengths from the duration model, while managing the schedule and constraints of each provider separately.

Handling No Shows

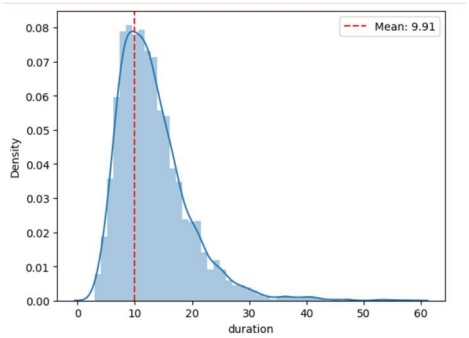
The scheduling model handles no shows by modifying the samples from the kernel density estimation with a binomial random variable representing whether clients show up for their appointments. The variable takes the value 0 with a probability equal to the no show rate and 1 with a probability equal to (1 - the no show rate). Then the expected value of the duration of the appointment is calculated using a value of the expected duration of the appointment if the clients show up and a duration of 0 if they are no shows.

Weighted Mean Absolute Error

To select an appointment start time, the model treats each possible start time as a potential forecast. The optimal forecast is the one that minimizes a metric known as the weighted mean absolute error relative to the possible appointment durations sampled from KDE.

Figure 2

Illustration of Weighted Mean Absolute Error



Intuitively, weighted mean absolute error compares the forecast to the possible appointment lengths, but does not treat over and underestimates of the duration in the same way. These two different circumstances are weighted differently because they have different effects in reality. If the model overestimates the duration of appointments, the appointments are completed before the next scheduled appointment, and the provider has idle time, but if the model underestimates the duration of the appointments, the appointments run over the scheduled time, causing clients to wait longer. Because the purpose of this project is primarily increasing utilization and decreasing makespan (the time it takes to complete all the scheduled tasks, which is a proxy for the time to first appointment for a new patient), the provider's unutilized time should be penalized much more than the patient's wait time. Weighted mean absolute error solves this issue by having a different weight for underestimates and overestimates. In this case, the weight is determined by a parameter α. This parameter represents the relative value of the providers' time to the clients' time. If α=2, the providers' time is essentially twice as important as the clients' time. Figure 2 shows an example of a forecast relative to the histogram of the appointment durations. In this case, the forecast is the mean, so it would be the optimal forecast when both the providers' and clients' time are weighted equally. If $\alpha < 1$, the providers' time is less important than the clients' time, so the optimal forecast would be to the right of the line in the figure, causing the providers to have more unutilized time, but reducing the potential time that clients would have to wait. If $\alpha > 1$, the providers' time is more important than the clients' time, so the optimal forecast would be to the left of the line in the figure, causing the providers to have higher utilization, but increasing patient wait times.

Weighted mean absolute error is represented by:

$$WMAE = \frac{1}{n}\sum_{i=1}^{n} w_i |y_i - \hat{y}|$$

$$w_i = \alpha \text{ if } y_i < \hat{y}, \text{ else } 1$$

where $\hat{y}$ is the predicted duration and y_i is the actual duration.

Aggregation of Appointment Durations

The start time of an appointment is chosen by first finding the appointments that occur before it on the particular day it is to be scheduled. Each of these appointments has a kernel density estimation of the probability distribution of possible durations, and these durations are sampled for each appointment. Sample durations are drawn from the probability distribution and modified as described above to account for no shows. The samples for each appointment earlier in the day to the appointment being scheduled are then aggregated through summation, leading to another histogram of possible times that the appointment in question should be scheduled. The start time for the appointment is then chosen by selecting the one that minimizes the weighted mean absolute error objective function.

Block Sizes

The model treats the minimum duration of appointment times as blocks. For example, in a 30/60 schedule, the block size is 30 minutes and appointments can take either one or two blocks. Smaller blocks, such as 5, 10, or even 1 can allow the model to schedule appointments more accurately by reducing the distance between the number of scheduled blocks and the time estimate durations from the model. For example, if the model outputs that an appointment should start at 9:12, the most accurate schedule a 30 minute block duration would be able to achieve would be either 9 or 9:30, but a 5 minute block schedule could have the appointment start at 9:10 or 9:15.

Output

For each appointment, the model finds the earliest time that it can be handled and schedules it for that time. The model reduces lost time due to the variance in appointment lengths and accounts for no shows. As an output it returns each appointment, the scheduled start time, and the provider that handles the appointment.

The Simulator

The simulator tests the schedule generated by that data against realized appointment durations. It achieves this by first generating the schedule on the appointments and their relevant data, and then testing how the schedule would have actually occurred by using the realized durations. No shows are generated randomly using a binomial variable and durations for these appointments are set to 0. The simulator creates a CSV file that shows the appointments, their scheduled start and end times, and their actual start and end times, and generates metrics, such as makespan and utilization that it puts in a separate file.

Sensitivity Analysis

We performed a sensitivity analysis on different parameters that affect the model's results. By varying values of α, the parameter used to control the relative importance of provider and patient time, and the block size, we captured the effects of these changes on the metrics in the simulation. We found that high values of α tend to increase patient wait times and decrease makespan, and low values increase providers' unutilized time and increase makespan. Since our objective is to minimize the makespan, this means that we should use a higher value of α. However, we do not want to increase patient wait times to an unendurable extent, so we need to optimize for the α that minimizes the makespan with a constraint that the mean wait time for clients does not exceed a certain amount.

Preliminary Results

We tested the scheduling model for 6 providers on 4000 appointments with a mean duration of about 13 minutes. As a baseline, we use a time slot duration of 18 minutes, which is similar to the sponsor's current utilization, and schedule one appointment for each time slot for every provider because this leads to a utilization of about 77%. The baseline model has a makespan of

26,385 working minutes. The scheduling model with $\alpha = 2.9$ and a time slot duration of 1 minute minimizes the makespan to 21,708 minutes and increases the utilization to 93.5% while ensuring that the average wait time for clients does not exceed 15 minutes. The decrease in makespan is about 17.7%, meaning that the working time in minutes to a new patient's first appointment is about 4/5 of the baseline model. Assuming that each provider sees appointments 4 hours a day twice a week, the time to a new patient being seen would be 3.8 months instead of 4.6 months and will allow seeing 21% more clients in a year. These results are an impressive improvement. However, they may be difficult to achieve in reality because they would require a full overhaul of the scheduling process. The model for predicting appointment durations would have to be developed and deployed, and appointments would require being scheduled at a minute granularity. The second constraint can be mitigated by increasing the block size to 10 minutes without a major loss in performance, as the model schedules multiple appointments for one time.

Model	**Utilization**	**Makespan (min)**	**Average wait time (min)**
Baseline	77%	26,385	5.4
Scheduling Engine α=2.9 block size=1	93.5%	21,708	14.7
Scheduling Engine α=2.5 block size=10	92.98%	21,926	14.4