DocPro

Team 11

eseseniordesignteam11@gmail.com

Mary Julianne Almazar

malmazar@seas.upenn.edu

Systems Science and Engineering

Karim El-Sewedy

karimels@wharton.upenn.edu

Systems Science and Engineering, Marketing and Operations Management, and Entrepreneurial Management

Miku Fujita

miku@seas.upenn.edu

Systems Science and Engineering

Swetha Subramaniam

swethas@sas.upenn.edu

Electrical Engineering

Advisor

Professor Rakesh Vohra

rvohra@seas.upenn.edu

Special Thanks to

Professor James Won

james.won@seas.upenn.edu

I. Table of Contents

II. Executive Summary

DocPro aims to decrease physician burnout and, thus, increase the efficiency and effectiveness of hospital care. Our goal is to create a product that utilizes a relationship between NASA-TLX data and physician and patient data collected from physicians at the Children's Hospital of Philadelphia (CHOP). In the last year, we have met with professors and medical professionals from the Penn Nursing School and CHOP, such as Dr. Amy Sawyer and Dr. James Won, who have given us insight on physician needs and innovative problem spaces. In addition, we have been granted access to real data from CHOP, which allowed us to further develop our idea and build our machine learning model to help physicians decrease the effects of high perceived workload.

We drew insight from these meetings in order to carefully design a backend that takes in patient characteristics, predicts the workload of each patient in real time, and assigns patients to physicians in a balanced manner every shift. This backend utilizes our skills in machine learning and integer programming. Finally, we designed a simple and intuitive frontend that will allow physicians and floor managers to easily view all of the information in one screen. This app is seamlessly integrated to our backend through data tables stored in Amazon Web Services. This integration will allow for secure, real-time learning and updates as the more and more hospitals adopt our technology.

Overall, DocPro (1) automatically balances the workload for physicians every shift, (2) provides transparency to physicians into predicted workload levels, and (3) enables hospitals to track physician workload levels. We are confident that our design provides a robust solution to the problems of burnout and efficiency within the hospital system.

III. Overview of Project

In multi-year research conducted by the Agency for Healthcare Research and Quality, there has been an observed rising prevalence of burnout among clinicians – with many studies reporting numbers above 50%. This has led to increasing concerns with the impact this has on access to care, patient safety and care quality. With regards to access to care, burnt-out physicians are much more likely to leave hospitals or even leave the practice overall, this can reduce many patients' continuity of care. Additionally, with regards to safety and quality, physicians are often suffering from impaired attention, memory and execution functions. This threatens patient safety and the care quality they receive, especially with physicians dealing with critical conditions. Beyond these detrimental patient-facing issues, this is also a multi-billion dollar issue. According to the Annals of Internal Medicine, physician turnover and reduced productivity costs the healthcare industry $4.6B every year. This is not to mention the losses incurred from burning out and losing top physician talent over time.

Overall, current hospital management systems have dealt with this problem poorly. Management considers only patient acuity in quantifying workload, despite the fact that acuity is often not an accurate portrayal of the workload put on physicians and other healthcare workers. While there are some band-aid human factors solutions that are sparsely being implemented to alleviate the issue of burn-out, the system must instead actively work to balance the workload of physicians through the patient makeup, as well as by providing increased transparency on daily workload. This is, in part, due to the fact that hospitals today do not have infrastructure in place to collect detailed data about physician workload. As a result, even if they sought to balance the workload of physicians, there is no backbone to a data-backed solution.

Seeking to tackle physician burnout at its core, DocPro has three main objectives in its bottom up approach to this issue: (1) balance workload for physicians, (2) provide transparency to physicians into predicted workload, and (3) enable hospitals to track physician workload levels. The DocPro system empowers physicians to self-report their own patient workload, based on the existing, well-researched NASA-TLX workload survey. Based on this information, it will create an optimally assigned balance of patient-related workload and will track over time which types of patients are contributing to workload in what way. This solution is particularly innovative as it is an integrated system that combines a front-end physician facing Web Application that feeds into and enriches the backend Machine Learning and Integer Programming infrastructure, thus creating a virtuous cycle of insights and assignment over time.

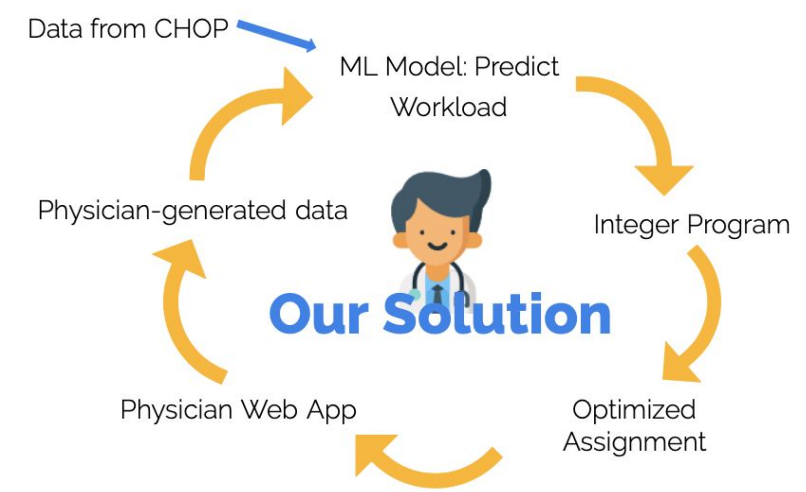

Figure 1
An overview of our solution from start (data from CHOP) to finish (physician-generated data). This figure not only highlights the individual parts of our solution but also emphasizes its cyclical nature.

In the regular healthcare landscape, the societal impact that DocPro seeks to have on physician burnout is an important one, both for the implications on physician job outlook and on the quality of patient care. In the current healthcare crisis, as a result of the coronavirus, managing physician workloads and maximizing output within the daily time constraints becomes even more important. With overstretched health facilities, it is pivotal to have infrastructure that balances workload of the frontline soldiers against COVID-19 to the extent that it is possible. While the DocPro system in its current form certainly would need to be modified to implement new features for times of crisis, it is clear that the objectives and landscape of this project are important and unfortunately timely.

Business Analysis

In multi-year research conducted by the Agency for Healthcare Research and Quality, there has been an observed rising prevalence of burnout among clinicians – with many studies reporting numbers above 50%. This has led to increasing concerns with the impact this has on access to care, patient safety and care quality. With regards to access to care, burnt-out physicians are much more likely to leave hospitals or even leave the practice overall, this can reduce many patients' continuity of care. Additionally, with regards to safety and quality, physicians are often suffering from impaired attention, memory and execution functions. This threatens patient safety and the care quality they receive, especially with physicians dealing with critical conditions. Beyond these detrimental patient-facing issues, this is also a multi-billion dollar issue. According to the Annals of Internal Medicine, physician turnover and reduced productivity costs the healthcare industry $4.6B every year. This is not to mention the losses incurred from burning out and losing top physician talent over time.

Overall, current hospital management systems have dealt with this problem poorly. Management considers only patient acuity in quantifying workload, despite the fact that acuity is often not an accurate portrayal of the workload put on physicians and other healthcare workers. While there are some band-aid human factors solutions that are sparsely being implemented to alleviate the issue of burn-out, the system must instead actively work to balance the workload of physicians through the patient makeup, as well as by providing increased transparency on daily workload. This is, in part, due to the fact that hospitals today do not have infrastructure in place to collect detailed data about physician workload. As a result, even if they sought to balance the workload of physicians, there is no backbone to a data-backed solution. This establishes the need, importance and timeliness for a solution like DocPro.

Stakeholders

While there are several stakeholders involved with any healthcare-related project or proposition, the main stakeholders here are physicians, patients and hospitals. The primary stakeholders involved are naturally the physicians, as they are on the front-line of the healthcare industry experiencing the fatigue and burnout. DocPro seeks to enhance the working experience and satisfaction levels of physicians generally, as well reduce reported burnout levels of course. To maximize the impact that DocPro has in alleviating this problem, the adoption and acceptance of the product by physicians is a key factor in its success. This is especially true since there is a physician-facing component to the DocPro system – that conveys the patient load for the day, predicted workload, and enables NASA-TLX data collection. Ultimately, the impact the product has on their experience on the job can be measured through metrics such as burnout, fatigue, satisfaction and overall turnover. Even though the product has been crafted with the physician in mind, if hospitals were to adopt the product and enforce it as policy amongst physicians across the entire health system, the physicians would likely have to adapt over time. With incoming and newly hired physicians, they would simply be onboarded to the job with DocPro in place, making the transition much more seamless. This leads to the secondary stakeholder of hospitals. Hospitals are arguably the most important stakeholder in the initial phases of launching and selling the product. This is because, even though we trial the service for free, physicians will only be able to interface with the product if hospitals are convinced of its merits and decide to implement or at least trial a novel system. Additionally, many hospitals operate with very antiquated, legacy systems and are averse to much technological change operationally, especially with such a large impact across the entire health management system. This is part of why the necessary data infrastructure to enable the balancing of workload that DocPro seeks to achieve is not prevalent. However, this increases the value of the overall system for intelligently building that capability in. Finally, the tertiary main stakeholder, but perhaps the most impacted socially is the end patient. As mentioned in the overview, the largest perpendicular problems associated with physician burnout is the quality of patient care, access to care and patient safety. Considering that is one of, if not, the main function of practicing physicians is to ensure the aforementioned characteristics of patient experiences, this is an important factor of understanding the impact on this stakeholder. Especially with the current healthcare crisis, access to patient care is an issue that members of this stakeholder group will pay increasing attention to when making related life decisions. This is a consideration that many in developing countries had dismissed for a long time, but is now top of mind once again. The COVID-19 crisis will unite these three stakeholders to ensure that solutions similar to what DocPro is offering become priority.

Value Proposition

The most structured way to assess the value proposition of DocPro is within the context of the aforementioned stakeholders. To start, with physicians, the value proposition is clear, since the entire system is inspired by and centered around accommodating physician needs. The product provides physicians with a greater sense of agency around contributing to balancing their workload and a greater sense of transparency in understanding the patient load on any given day. Overall, DocPro will result in increased productivity and satisfaction, while reducing the perception of workload and feelings of burnout. This is a key value proposition, as our hypothesis is that it will also lead to a decreased rate of physician turnover. Turnover and productivity are important considerations for the monetary-value providing stakeholder: hospitals. In actuality, DocPro’s customer is hospitals, and in many cases, private hospitals especially will realistically only implement a new system that comes with this product if it provides them with a clear monetary value. The reality is that physician turnover is a serious financial problem, specifically because a hospital can spend at least $27,000 on each new physician they hire, just in recruitment costs, and the physician turnover costs are about $400,000-$600,000 per physician (due to retraining and other onboarding costs). With the current staggering turnover rates, the average hospital is losing $5.595M as a result of this systemic issue. The value to hospitals is clear in this regard, but a secondary source of value is the high and consistent quality of patient care and safety, both in reputation and actuality. This is important, because the quality of patient care directly contributes to the prestige and standing of a hospital, especially when it reaches a certain echelon within the healthcare network. This leads to the value proposition of the tertiary stakeholder: patients. With patients, the value primarily comes from improved service, quality and safety associated with the care they are receiving from physicians, which is sure to have an even broader impact on other employees in the hospital, such as nurses, as well. While this value is within each interaction, over many interactions, this reduced burnout and turnover results in greater access to care, which is an even more precarious concern with the coronavirus crisis. It is important to note the reality that the majority of Americans do not maintain much optionality when it comes to hospitals and where they choose to go. It is much more based on convenience and accessibility, but it will be interesting to assess how this changes in the world after coronavirus.

Market

While the original inspiration for DocPro came from secondary research and personal experience, the vision for the system and the need that the product serves was heavily refined through a combination of primary and further secondary research. The main goal of this process was to better understand the overall problem space and figure out how different stakeholders, specifically at the University of Pennsylvania Health System, would react to the solution we were seeking to provide. By talking to a variety of physicians, professors and administrators, we understood that any full-packaged solution needed to alleviate the foundation of the workload management problem: lack of data infrastructure to collect detailed information on physician-patient specific workload. With this insight underpinning what we sought to accomplish, through further conversations with physicians at larger hospitals and small clinics, we honed the market opportunity for the problem area. After understanding that smaller clinics would be served to a less valuable extent by the DocPro system, since network effects are realized to a much smaller extent, the primary target market was deemed to be hospitals. This market comprises hospitals of all sizes, both those experiencing existing systemic physician burnout and turnover problems as well as those looking to implement preventive measures and maintain a generally high quality of care.

To provide deeper insight in this choice of market opportunity, our hypothesis is that the customer segment will be hospitals rather than physicians directly. Even though the active, daily pain of burnout is felt by physicians, in the outlined customer segment, physicians exist within and are managed by these broader healthcare systems. As a result, they utilize the software that their respective institutions prescribe rather than of their own volution. To sell this technology, it is also more logical to take a top down approach in having the software trickle down from administrators, rather than try to go B2C in selling to individual physicians. Since DocPro relies on patient data to feed into the machine learning and integer program, the value would not be at its maximum if hospitals were not on board. Even more importantly, while physicians are experiencing reduced productivity and feelings of fatigue and burnout, they often cannot quantify the economic value of that loss. However, to an entire hospital, they can directly see the costs associated with onboarding, retraining and reduced productivity, which incentivizes them as a customer to purchase access to the SaaS product. This is not to say that it would not be effective to market this product to physicians as well, as it would be incredibly valuable to have them serve as ambassadors and salespeople of sorts, on the inside, while the business development team sells them on the outside.

Market Size
One of the main goals of this process was to estimate the size of the overall market. To identify the overall value of this business proposition, understanding the total addressable market, specifically in the US to start, is essential. This market sizing was conducted by researching the number of hospitals, estimating the average price spent yearly on this product, and taking a conservative estimate of the churn rate for a SaaS product of 7%. This entire market sizing was conducted very conservatively, which was important considering DocPro will be used to serve hospitals hoping to take preventive measures, as well as those severely suffering from existing physician burnout issues. This resulted in an overall estimated market size for this market of $1.155B. While this segment of the healthcare industry is fairly stable, especially for such the consistent need of serving physicians and managing physician workload, DocPro expects that this market opportunity will grow in line with the overall healthcare industry. This is an estimated growth rate of 2.5-2.8% for the entire market opportunity. Beyond aligning with the rest of the hospital industry, this is also aligned with the growth rate for the number of physicians in the US, which is growing at an annualized rate of 2.2%. This number is expected to grow beyond this rate with the deficiencies that the COVID-19 crisis has exposed.

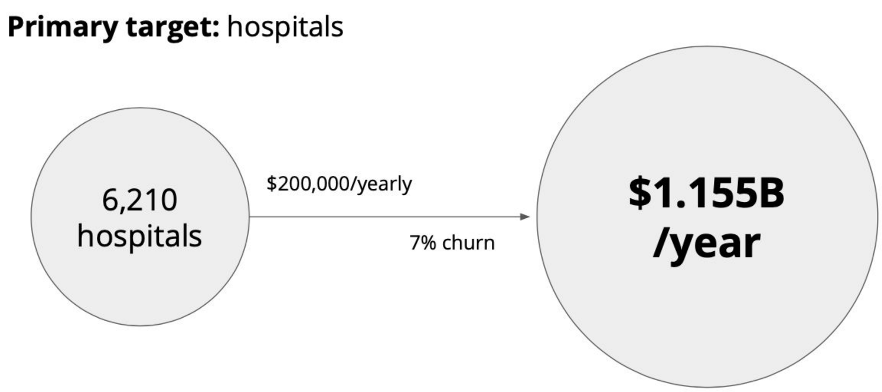

Figure 2
An overview of our market size.

Revenue Model
To better understand the above market analysis, it is important to dive deeper into the revenue and cost model for DocPro. As mentioned, DocPro is designed and developed to be sold as a SaaS product, which is essentially software hosted on cloud infrastructure and where companies can pay a monthly or yearly subscription fee to gain access to the product. SaaS products alleviate the need for initial investment into creating IT infrastructure to support the software, which removes barriers for customers to trial the product. To determine a preliminary estimate for price, which is the foundation for the market sizing, the yearly economic value provided to hospitals was reverse engineered. By combining multiple studies, we found that on average, physician turnover costs at least $400,000 and likely closer to $600,000 per physician. Considering the current physician turnover rate is 8% across 1.085M physicians in the US, the average cost of physician turnover across the 6,210 hospitals in the US is $5.595M per year. This does not take into account costs associated with reduced productivity, which are estimated

to be over $27,000 per physician in the US. We then took a very conservative approach, by essentially splitting that value by 10 at the highest end price, based on the tiering system below.

Tiered pricing based on number of physicians
- Large size (1,000+ physicians): $60,000/month
- Medium size (500-1,000 physicians): $30,000/month
- Small size (100-500 physicians): $10,000/month
- Local size (1-100 physicians): $5,000/month

Figure 3
An overview of our tiered pricing model.

This tiered pricing system is based on the number of physicians per hospital, which is a metric that is common to hospitals. The reason we decided to create tiers, as opposed to charging hospitals by physician, is to incentivize hospitals to onboard its entire employee population and not limit the usability of DocPro to specific departments. However, since business development and account managers will likely be working with clients in a personalized manner, the pricing structure can be catered to each hospital's evolving needs. While this is an initial estimate or price, this is a conservative estimate considering the tremendous cost savings DocPro provides, not even including the savings accrued as a result of the separate data collection system for physician workload.

Costs
Since the entire DocPro system has yet to be fully implemented at a hospital, it is tough to estimate exactly the costs of acquiring one hospital as a customer and scaling those up to the entire market. However, research into similar systems have shown that our main expenses will be cloud expenses, specifically AWS, sales expenses, which will be fairly high since the cost of customer acquisition for such large customers tend to be on the richer end of the spectrum, and, of course, business and operations expenses. Even though it is tough to accurately estimate the sum of the operational costs, we believe that to run an effective MVP at a health system as sophisticated as CHOP's, which would be our first step as a business, it will cost around $10K, primarily consisting of cloud expenses. However, for the purposes of senior design, we spent $0 in total, since all our engineering costs were covered by the in-house talent of the team.

Competition
The team's confidence in the product's potential to capture a sizable portion of the market comes from the fact that there are competitors that directly rival the integrated and packaged nature of DocPro. Our closest category of competitors would be hospital inpatient Electronic Health Record (EHR) systems. This market continues to be dominated by Epic and Cerner. Cerner has found success in the public sector, while Epic has the lion's-share of the private sector. At a high-level, an EHR is a collection of patient and population health information that is electronically-stored in a digital format. While EHR systems can include a wide range of data, their claim to provide value in the form of increasing quality of care falls flat in comparison to the workload-focused system of DocPro. The main difference is that EHR providers are facilitating

care management programs, which help increase quality of care by preventing hospitalizations among high-risk patients. This is essentially decreasing the frequency of exposure to health care, not necessarily ensuring a higher quality of care. DocPro's differentiation comes from its approach to tackling the problem of physician burnout and turnover to push for higher quality of care and patient safety in every healthcare interaction. In this pursuit, DocPro does not have any direct competitors. In many cases, what EHR providers are attempting to do are band-aid solutions that do not solve the problem at its foundation. That being said, given the widespread prevalence of EHR systems, DocPro plans on exploring a mechanism to modularize DocPro to be integrated and compatible with the biggest players in this market. The hope is that this will alleviate any frictions to adoption and perhaps be able to partner with EHR systems to upsell hospitals with the DocPro solution.

IV. Technical Description

When designing the solution, there are several requirements and constraints that must be considered. First, in terms of requirements, the solution must be usable by physicians on a hospital floor, meaning that the software that completes the assignment and tracks workload must be user-friendly and able to be understood by a person with a non-technical background. Thus, we decided to create a dashboard with a simple UI that has the assignment programming and machine learning embedded into it but which is not visible to the user. The dashboard will also show the assignment in a way that physicians can understand as opposed to the binary assignment matrix that will be given by the optimized assignment program. Second, in terms of requirements, the workload must be balanced between physicians, such that no physicians have a workload that is significantly higher than the others. This requirement of balancing the workload is included as the objective function of the mathematical program that is used to find the optimal assignment. Thirdly, the patient data must be stored such that the machine learning algorithm can learn which types of patients are giving physicians higher or lower workload. Next, in terms of constraints, the solution cannot require more physicians than are currently on the floor or fewer patients than are currently on the floor. The solution being designed is purely one that assigns patients to physicians every shift, and more physicians cannot be added once the shift begins. Similarly, patients cannot be transferred to other hospitals or departments once the shift begins. Thus, the solution must balance workload given the current physicians and patients that are previously given. The last constraint is the data privacy constraint. Because of the Health Insurance Portability and Accountability Act (HIPAA), personally identifiable health information cannot be used or stored in the design. Thus, all information being used must be anonymized before being analyzed.

A few iterations of the solution have been considered. The first iteration of the design focused on minimizing perceived workload of nurses. This iteration assumed that different types of patients contributed differently to different nurses. For example, one nurse may feel less workload from a patient who is more complex compared to one who is more unstable, while another nurse may feel the opposite. The five patient factors that were considered were stability, complexity, predictability, resiliency, and vulnerability. These patient factors would be recorded along with nurses' preference between these patient factors, and the assignment would be optimized based on these preferences. The second iteration of the design was focused on balancing workload across different nurses, since experts in the nursing field told us that the workload impacts of these patient factors were small and not considered to be impactful by nurses. This iteration allowed a nurse to label patients as “light,” “medium,” or “heavy” workload, which a machine learning algorithm would track in order to learn which types of patients contributed heavily to workload. However, because of patient privacy laws that prohibited us from collecting this kind of data without Institutional Review Board approval, we were not able to proceed with this design. Thus, we partnered with Professors James Won and the Children’s Hospital of Philadelphia, who were able to give us access to deidentified patient and physician workload data. We shifted our focus from nurses to physicians, since the data available to us was based on physicians, and nurses and physicians deal with very similar workload issues. This final iteration of the solution focuses on NASA-TLX workload scores. NASA-TLX is a widely used, multidimensional assessment tool developed by NASA that rates perceived workload. It measures workload on the dimensions of mental demand, physical demand, temporal demand, performance, effort, and frustration. Our solution focuses specifically on predicting and assigning based on temporal workload, as we were told by various hospital staff, including Ms. Rebecca Love of HireNurses.com and Ms. Kelly Meyers of CHOP’s oncology department, that the time pressure of the task was the most important feature of workload in a hospital. Thus, the current approach aims to create a dashboard to assign patients to physicians based on the NASA-TLX defined temporal workload of patients.

In this design, past workload data that has been collected is used to develop a machine learning algorithm that will be able to create initial predictions of the workload of patients. The data we were working with had two parts: the physician workload data and the patient characteristics data. The physician workload data included the timestamp of the survey and the NASA-TLX scores (both per dimension and total). The patient characteristics data had the hospital admission date and the following characteristics: age, sex, gestational age, hospital length of stay, oncology length of stay, inpatient length of stay, ICU length of stay, whether the patient had a complex chronic condition, whether the patient was medically complex, whether the patient was in the ICU, whether the patient was in the emergency department, whether the patient was an elective or emergent admission. Since there currently was no tool to identify which physicians treated which patients, we decided to assume that physicians who took the survey on a particular day treated the patients who were admitted on that day. Thus, the workload data was collapsed such that each day had an average score for each dimension. Then, the two datasets were joined based on the date of the timestamp. An additional variable was added to track whether the day was a weekend or a weekday. Thus, each patient now had associated workload scores based on the average workload of the physicians on the floor that day.

Based on this dataset, a machine learning algorithm was implemented to predict temporal workload. While a random forest regression was initially implemented, we found that a decision tree regression was a better fit for the small dataset of around 100 points that we were working with. Thus, using hyperparameter tuning of four parameters, we achieved an average cross-validated r-squared value of 0.31. While this accuracy is lower than is ideal, we hope that the current algorithm will improve as the system is in place and more physicians take the workload survey embedded in the application. The current decision tree algorithm implemented in Python is attached in Appendix 1.

The second part of this system is the assignment aspect. Once the machine learning algorithm has been implemented successfully, it will be able to predict the temporal workload of each new patient. Based on this, the system will assign patients to physicians for each shift. The assignment problem is modeled as a mathematical program, where the objective is to balance workload between physicians. Below is the technical description of the mathematical program:

$Define\ constant\ P\ as\ number\ of\ patients$

$Define\ constant\ D\ as\ number\ of\ physicians$

$Define\ cost\ matrix\ w_{pd}\ with\ P\ \ rows\ and\ D\ columns, which\ includes\ the\ workload\ of\ each\ patient.$

$The\ workload\ of\ each\ patient\ is\ equal\ to\ their\ predicted\ temporal\ workload.$

$Define\ assignment\ variable\ a_{pd},\ which\ takes\ a\ value\ of\ 1\ if\ patient\ p\ is\ assigned\ to\ physician\ d,\ and\ a\ value\ of\ 0\ if\ patient\ p\ is\ not\ assigned\ to\ physician\ d$

$Defined\ below\ is\ the\ mathematical\ optimization\ model:$

$$Minimize\ \sum_{i=1}^{D}\sum_{j=1}^{D}\left(\left|\sum_{p=1}^{P} w_{pi}a_{pi} - \sum_{p=1}^{P} w_{pi}a_{pj}\right|\right)$$

$Subject\ to$

$$\sum_{d=1}^{D} a_{pd} = 1\ \forall\ p \in P$$

$$a_{pd} \in (0,\ 1)$$

The objective function of this model minimizes the difference in total workload between every pair of nurses. The first constraint ensures that each patient has one nurse. The second constraint defines this model as an integer program. This integer program will give a solution that assigns every patient to a physician. After this mathematical program runs within the dashboard, the dashboard shows the assignment as each physician is given different patients. The integer program implemented in Python is attached in Appendix 2.

We decided to use Amazon Web Services (AWS) to create a serverless backend for our web application. Using a serverless backend allows us more flexibility, scalability, and a more affordable pay-as-you-go pricing plan as compared to a traditional server infrastructure. As students, we were able to obtain credits that gave us access to these resources for free. There are several advantages to using AWS, but the most important is security. Given that we are working with patient information, though de-identified, it is crucial that we protect this data by storing it on a secure platform.

First, we uploaded the two datasets received from CHOP into buckets on Amazon's Simple Storage Device (S3). This is an easy way to store large volumes of data. Next, we used an AWS Lambda Function to push the data from the buckets into tables on DynamoDB called Patient and Workload. DynamoDB is a robust NoSQL key-value store with built in security and automatic backup. Once the tables were populated, another Lambda Function was used to connect the tables to the frontend web application, enabling data to be uploaded to the cloud in real time. This means that our application can accept synchronous data through the front end web application as well as data files dumped into buckets on the back end. By choosing AWS S3 and DynamoDB, we ensured that regardless of the file type, size, or quantity, our application can seamlessly intake and process the datasets required to operate successfully.

This data was then displayed on our web app. We carefully designed our frontend to provide a simple and intuitive process for our users. The main screen shows physicians all the information they would need for their shift. First, the heading displays the date, shift type, and shift time. Then, the physician-to-patient assignment is clearly displayed in the middle of the screen. Underneath, the user can find two lists: the active physician on shift and the admitted patients. Finally, a navigation bar at the bottom of the screen holds three important buttons: add physician, get result, and add patient. This navigation bar is uncluttered and clearly states the actions that the user can perform on the web app. This sentiment of simplicity and intuitiveness is applied to all other screens of the web app. Images of our web app are attached in Appendix 3.

The web app also allows the physicians to easily take a workload survey at the end of each of their shifts. A notification will pop up as their shift ends and ask them whether or not they would like to take the survey. Upon taking the survey, the app will update the physician's information throughout the entire system. Furthermore, to emphasize our intention to bring ease to physicians, the web app will send physicians notifications when their workload is particularly high. This will not only flag the current shift as a data point for the machine learning algorithm but also prepare the physician for a relatively harder shift.

As the dashboard develops, the machine learning algorithm will be able to make better predictions on the temporal workload contributions of a new patient. In addition, as the system is used over a longer period of time, hospital management will have access to more robust data on workload, allowing them to see when higher spikes in workload occur so that those shifts can be appropriately staffed.

The final status of the project is an integrated dashboard with a machine learning algorithm to predict patient workload, an integer program to assign patients to physicians, and a user-friendly UI that allows physicians to easily take the workload survey for continuous improvement of the workload predictions. Test results of the system show promising results. As mentioned previously, the decision tree algorithm achieved a cross-validated r-squared of 0.31. Test results of the optimization program using simulated data show that balanced workload can be achieved through the current model. We were unfortunately unable to get substantive quantitative test results since testing the system at CHOP was not feasible given the busy schedule of the physicians, the controls needed in the experiment, and the patient privacy concerns.

In summary, the technical aspects of the project include a machine learning algorithm that predicts patient workload, an integer programming model that assigns patients to physicians to balance workload, and a dashboard with user-friendly UX that shows the predicted workload and assignment and also allows the physicians to take the workload survey. Ultimately, the goal of this project is to (1) automatically balance the workload for physicians every shift, (2) provide transparency to physicians into predicted workload levels, and (3) enable hospitals to track physician workload levels.

V. Self Learning

Before honing in on the specific focus of our project and the angle from which we wanted to approach this problem, we took it upon ourselves to research previous literature on the problem of physician burnout and fatigue as well as the broader healthcare industry. Using a combination of Penn's internal research resources and press search (i.e., Google), we learned about previous attempts to tackle this issue, which was invaluable in getting us started. We built upon this knowledge with qualitative primary research, which we conducted by discussing with academic and industry experts as well as physicians and nurses themselves. Some of the most informational conversations came from speaking with Professor James Won, who works with the ESE department as well as the CHOP Research Institute, Ms. Rebecca Love, nurse and founder of HireNurses.com, Dr. Marion Leary, the Director of Innovation at CHOP, and more. These individuals were invaluable to us in narrowing the focus of our problem, obtaining data, and providing feedback in between iterations of the project.

From a more technical skill lens, we have integrated several technologies, including Integer Programming, Machine Learning, Python, Javascript, and Amazon Web Services. Given our various experiences both in classes at Penn and during internships, at least one member of our team was highly skilled at using each tool. Of course, there were certain times throughout the semester where we had to apply the tools in a manner that was new to us, but this was a welcome challenge, which we mostly used online resources to conquer. Additionally, we were lucky to have the support of the Senior Design TA team for advice regarding any obstacles we faced. The fact that there was a fairly skilled member of the team in each tool enabled a learning opportunity for all the other members of the team, as we sought to work on most parts of the project together. The academic foundation of our project really came from the classes we have taken over the past three and a half years. While they span a few departments, the most prominent was unsurprisingly ESE. While there must be more classes that would have expanded the scope of the project, the most helpful ones were ESE305 (Foundations of Data Science), ESE204 (Decision Models), CIS419 (Applied Machine Learning), NETS212 (Scalable Cloud Computing), ESE407 (Introduction to Networks and Protocols), and HCMG250 (Healthcare Reform).

VI. Ethical and Professional Responsibilities

This project fits seamlessly in the context of the American healthcare system and the various inefficiencies and problems that undeniably plague it. Along with different parts of the healthcare value chain lie issues with both the way the industry functions internally and the care that the American population receives externally. Policy makers and influential industry players alike are beginning to more seriously acknowledge these problems and take action towards them. This sets up societal and political context for this project fairly well, as there is a greater prioritization of quality and lower cost. The fact that physicians are experiencing such high rates of burnout and fatigue makes this problem ever more pressing.

In a world where data privacy and security have become a monumental concern, trying to build data-driven products, especially in the realm of healthcare, is increasingly difficult. For this reason, we spent many months and meetings searching for an accessible dataset to develop our model. The data we obtained from CHOP was completely deidentified due to HIPAA restrictions; the only variables involving personal information we were given access to were patient age and sex. We considered security to be a highly important feature of our application. By separating the user facing application from the server side, we guarantee that the sensitive data cannot be accessed directly by any user, and that the data will be saved to the database with a strong guarantee. Additionally, AWS has security features built into their systems that ensure reliability and consistency.

VII. Meetings

Our team met with our advisor, Professor Rakesh Vohra, three times during the initial stages of the project to ensure that we were on the right track and to affirm that the project idea was a viable one. We decided to limit it to a few meetings at first, as Professor Vohra connected us with many other people to liaise with regarding our project.

After these initial meetings, we tried to meet with Professor Vohra on a regular bi-monthly basis. In these meetings, our goal was to update Professor Vohra on any progress we made on our project and gain any helpful insight he might provide. In addition to this, we provided Professor Vohra with copies of our work, such as our posters and executive summaries, via email throughout the year.

Towards the end of the year we communicated with Professor Vohra frequently despite extenuating circumstances. These meetings were focused on polishing our finished product from our presentation slides to the tone of our pitch. Overall, Professor Vohra has been very helpful, accessible, and accommodating. We thank him for all the time he has dedicated to us.

VIII. Proposed Schedule with Milestones

By the end of the fall semester, we successfully narrowed our project focus, did ample research on the topic by speaking to several professionals, and built a rudimentary model with simulated data. For the spring semester, we had three major milestones:

1. Improve modelling with real data
2. Build the user web application

3. Finalize value proposition and business plan

We are proud to state that we have met these three milestones successfully. We obtained data from the Children's Hospital of Pennsylvania with the help of Dr. James Won. Using this data, we finalized our integer programming model and constructed our machine learning model. With the knowledge and advice from the healthcare professionals we consulted, we then designed a user interface we believed would be practical and convenient to display the result of our data-driven optimization. Finally, we proposed that our product, DocPro, could be especially useful in research on physician workload that may be informative in situations like the Coronavirus pandemic, when hospitals are at full capacity and the whole staff is overworked. We hope that DocPro will help us understand physician workload balancing and perhaps help ease the burden on doctors, nurses, and healthcare professionals working nonstop in a time of crisis.

IX. Discussion of Teamwork

Our team usually meets in person or virtually to work on the project and complete assigned presentations, reports, and videos. We typically divide the work as equally as possible, and nearly every aspect of the project has been a joint effort from each member of the team. The remaining work is naturally divided by expertise: Miku has background knowledge on the subjects of physicians and healthcare, so she usually engaged in discussion with the hospital professionals and faculty to gather information. Miku also has a background in mathematical optimization, so she worked on the integer program. Miku and Swetha both have experience with machine learning, so they both developed the decision tree model. Swetha has experience with data storage, so she worked with AWS. Jules has the most expertise in design and web programming, so she worked on creating the frontend of the dashboard. Finally, Karim took charge of designing our business proposal and framing the marketability of our project as a real product.

X. Budget and Justification

Our team planned to complete our product with a budget of $0. This has been our budget since the beginning of the semester, and we were successful in completing our project within this limit. Our product consists of machine learning Python code and a dashboard created with JavaScript, which are all free technologies. We were able to store our data on Amazon Web Services through pooling our free student credit.

XI. Standards and Compliance

DocPro makes use of CHOP datasets, AWS, Python, and Javascript. Since we are working with patient data, our product must be HIPAA compliant. This means that we must adhere to the HIPAA Privacy Rule, HIPAA Security Rule, and HIPAA Breach Notification Rule. We abide by the Privacy Rule since our data is completely anonymized. We comply with the Security Rule since we are securely maintaining data in AWS. In the event of an unforeseen data breach, DocPro agrees to notify relevant parties.

In complying with HIPAA, DocPro also adheres to the IEEE P7002 standard, which reads as follows: “This standard defines requirements for a systems/software engineering process for privacy oriented considerations regarding products, services, and systems utilizing employee, customer or other external user's personal data.”

XII. Work Done Since Last Semester

This semester, the DocPro team obtained physician and patient datasets from CHOP, finalized the integer programming and machine learning models, built a user-facing web application, and determined an appropriate business plan for the final product.

XIII. Discussion and Conclusion

In summary, DocPro is a data-based solution to the problem of physician burnout. The team designed this physician workload dashboard in order to (1) automatically balance the workload for physicians every shift, (2) provide transparency to physicians into predicted workload levels, and (3) enable hospitals to track physician workload levels. Specifically, throughout this past year, we developed a machine learning algorithm to predict workload, created an integer program to assign patients to patients, and designed a user friendly application in which physicians can see their assigned patients, see their predicted workload, and continue to track their workload through taking the workload survey in the app.

Through this process, we have learned about the design process and the importance of industry research. Making sure that the project we were building was solving a relevant problem to the stakeholders was incredibly important. We first completed a literature review to learn about the problems within healthcare and the issue of burnout, and to understand the different approaches academics have considered in solving the assignment problem and quantifying workload. We verified and developed the project concept by speaking to over fifteen healthcare and hospital experts over email, on the phone, or in-person. Without the feedback of the experts that we spoke to, we would not have understood this problem and the needs of hospital staff as well, and our design would not have been as precise or accurate.

XIV. Appendices

Appendix I

```
reg = tree.DecisionTreeRegressor(max_depth = 1, min_samples_split=3,
                                 min_samples_leaf = 2, max_features = 13)
reg.fit(X, y)

cv_method = KFold(n_splits=10,shuffle=True)
R2 = np.mean(cross_val_score(reg, X, y, cv=cv_method, scoring='r2'))
print("Avg. R-squared: ", R2)

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20,
                                                    random_state=42)
reg.fit(X_train, y_train);
predictions = reg.predict(X_test)
errors = abs(np.round(predictions) - y_test)
print('Mean Absolute Error:', np.round(np.mean(errors), 3))
```

```
Avg. R-squared:  0.31071498914897094
Mean Absolute Error: 3.327
```

Appendix II

```
#Variables
vars = {}
for d in range(1, num_doctors+1):
  vars[d] = {};
  for p in range(0, num_patients):
    vars[d][p] = LpVariable("D{0}P{1}".format(d, p), 0, 1, LpInteger
```

```
#Objective Function
prob += lpSum(lpSum([lpSum(workload[d][p]*vars[d][p] for p in range(1, num_patients))-...
                     lpSum(workload[e][p]*vars[e][p] for p in range (1, num_patients)) ...
                     for d in range(1, num_doctors) for e in range(1, num_doctors)]))
```

```
#Each patients is assigned one physician
for p in range(0, num_patients):
    prob += lpSum(vars[d][p] for d in range(1, num_doctors+1)) == 1, "Patient Requirement {0}".fo
```

Appendix III