Predicting Degenerative Disc Disease in Chiropractic Patients Via Machine Learning

by

Dillon Harding

A Thesis Presented to

The Faculty of the Department of Information Technology

Illinois State University

Supervised By: Dr. Shaoen Wu

December 2021

Bachelor Honors Thesis in Information Technology

Title: Predicting Degenerative Disc Disease in Chiropractic Patients Via Machine Learning
Author: Dillon Harding
Faculty Supervisor: Dr. Shaoen Wu
Date: December 2021

Key Terms: *Artificial Intelligence, Chiropractic, Degenerative Disc, Machine Learning, Medical Diagnosis*

Abstract: While a majority of the medical field has seen machine learning introduced in some capacity, the chiropractic field has lagged behind other medical branches. A major application of machine learning in the medical field is producing predictive diagnoses of particular ailments. While chiropractors mainly focus on pain relief therapy, a condition commonly assessed by chiropractors is degenerative disc disease. X-ray imaging can be used to confirm the diagnosis of degenerative disc disease, but many chiropractors are able to discern degenerative disc disease based on patient metrics alone. Given this ability in medical professionals, machine learning models may also be able to assist in identifying degenerative disc disease, allowing for treatment to begin earlier.

Purpose: The purpose of this thesis is to research the possibility of using machine learning models in diagnosing degenerative disc disease. It aims to provide accurate diagnosis predictions based upon patient metrics and classifications from medical professionals such as chiropractors.

Method: This deliverable of this thesis was produced using machine learning models implemented in Jupyter Notebook and Spyder IDE. Data collection information was read using Microsoft Excel's CSV reading capabilities.

Acknowledgments

I would like to thank every individual who participated even in the slightest towards making this thesis possible. Your support and contributions are cherished and will always be remembered.

I would like to specifically thank Dr. Lori Wight for her amazing willingness to contribute to this thesis. Without her knowledge and decades of practice, the machine learning models produced would be starved of data.

Also, I would like to thank Dr. Shaoen Wu and Noah Ziems for their support throughout the conception and creation of this thesis. Noah Ziems and Dr. Wu are incredibly knowledgeable individuals whose assistance allowed for multiple machine learning models to be implemented.

With utmost gratitude,

Dillon Harding

Dillon Harding

Illinois State University, 2021

Table of Contents

TABLE OF CONTENTS (*Continued*)

Introduction

0.1 Thesis Aims

This thesis has three main aims regarding degenerative disc disease and machine learning:

1.) Provide a proper background to both machine learning and degenerative disc disease such as to create a basis of understanding for the project portion of the thesis. Real-world examples paired with relevant figures will be provided to help in concept comprehension.

2.) Present data in a manner that is comprehensible to an extent which readers of this thesis can understand conclusions made by machine learning models. Discussion of relevant sections of collected data will be necessary in order to show the correlations a machine learning model may make.

3.) Demonstrate the construction, results, and methodology of all machine learning models created for the purpose of this thesis. Description of the machine learning models should allow for general understanding from those in fields unrelated to machine learning. Though, the intricacies of the models will be explored to allow for future work to be done by those in the machine learning field.

0.2 Structure of Thesis

The structure of this thesis will progress in a series of sections as follows:

- **Background** will discuss information regarding both machine learning and degenerative disc disease. The machine learning portion will discuss the basic concepts of machine learning as well as its implementations seen in the medical field.

- **Data Collection** contains the methods used in obtaining the data on which all produced machine learning models were trained. Due to the sensitive nature of some of the data, specific agreements, as well as collection techniques, will be disclosed for transparency purposes.

- **Machine Learning Models** introduces all created machine learning models and the algorithms they were based upon. The benefits and drawbacks of each model as well as their respective performance will be analyzed. This section will contain the majority of information used in the conclusion of this thesis.

- **Conclusion** will explore the results observed within each produced machine learning model in conjunction with their usage implications.

- **Discussion** will address all insights and empirical findings throughout the progression of this thesis. The viability of the specific issue this thesis attempts to address will be compared to testimonies from professionals in the chiropractic field.

Background

1.1 Overview of Machine Learning

Throughout history, technology has been used to uncover breakthroughs in human understanding. One such revolutionary technology was the modern computer, created by Alan Turing in 1950 (Turing, 1950). A mere 6 years after the creation of the modern computer, the concept of artificial intelligence would be formally introduced by scientist and engineer John McCarthy (McCarthy, 2004). McCarthy defined artificial intelligence as "...the science and engineering of making intelligent machines, especially intelligent computer programs...related to the similar task of using computers to understand human intelligence" (McCarthy, 2004). Just as the Industrial Revolution allowed for the automation of manual labor, artificial intelligence has allowed for the automation of intellectual labor.

One heavily studied branch of artificial intelligence is machine learning. Machine learning is based on the manner in which humans learn, iteratively utilizing data sets and algorithms to increase understanding of a subject (IBM Cloud Education, 2020). By using statistical methods to train algorithms, machine learning can perform classifications and predictions on sets of data. Machine learning varies in complexity, from deep learning to classical learning (IBM Cloud Education, 2020). While machine and deep learning are occasionally used interchangeably, the two vary in their implementation and capabilities (IBM Cloud Education, 2020). Deep learning uses neural networks, modeled after human neurons, to learn adaptively, while classical machine learning models require manual changes to shift their learning capabilities (IBM Cloud Education, 2020). Manual changes, in this context, could be anything regarding changes to algorithms used, data included, or the programming structure of a machine learning model.

Machine learning models can be trained using one of two methods, supervised and unsupervised learning (Delua, 2021). Supervised learning refers to datasets in which classification labels are provided for each row of data. Such labels allow for the measurement of a model's accuracy by dividing the total data set into respective training and testing sets (Delua, 2021). Training and testing data sets often observe an 80/20 split, though other ratios like 70/30 are also viable (Delua, 2021). On the other hand, unsupervised learning allows for models to perform analysis where classification labels are not present (Delua, 2021). While supervised learning is typically used for classification problems, unsupervised learning is better suited for association problems. By calculating the similarities and differences of data in a data set, an unsupervised learning model might create separated clusters of similar data (Delua, 2021).

In supervised learning models, there are many ways to measure the accuracy of predictions made on a test data set. Of course, one of the most seemingly obvious methods would be a simple test of accuracy, the percentage of predictions that were correct versus those that were not. This test can be performed by analyzing a chart of actual classifications compared to the model's classifications. (See **Figure 1**).

Figure 1

		Predicted/Classified	
		Negative	**Positive**
Actual	**Negative**	998	0
	Positive	1	1

Example of Classification Data. Reprinted from Medium, by K. P. Shung, 2020, https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9

While accuracy may seem like an excellent method of progress measurement, it does come with potential risks. Referencing **Figure 1** above, data scientist Koo Ping Shung (2020) begs the question "What if the positive here is actually someone who is sick and carrying a virus that can spread quickly? Or the positive here represents [a terrorist] that the model says is a non-terrorist?". To address this issue, other methods of measurement such as precision, recall, and F1 score can be used as compliments (Shung, 2020). Recall and Precision are calculated using the formulas depicted in **Figure 2** below.

Figure 2

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Formulas for Precision and Recall. Reprinted from Medium, by K. P. Shung, 2020, https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9

Precision measures the number of true positives classified by a machine learning model against the total number of positives classified by the model. Recall on the other hand measures the number of true positives classified by a machine learning model against the actual number of positives in the data set. Precision works well in scenarios where guaranteed positives matter, such as when making an investment (Santos, 2020). Recall, on the other hand, is important where false negatives are critical, like detecting faulty equipment (Santos, 2020). F1 score is yet another method of measurement which is a function of both recall and precision. The formula for F1 score can be seen in **Figure 3** below.

Figure 3

$$F1 = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

F1 Score Formula. Reprinted from Medium, by K. P. Shung, 2020, https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9

Machine learning finds applications in nearly every industry and study around the world (Deo, 2015). Data is a valuable resource in machine learning, as it is necessary to achieve properly trained machine learning models. As such, machine learning is seen most predominantly in fields that produce large amounts of data (Deo, 2015). For instance, the healthcare industry, customer service chatbots, and the popular Apple companion Siri are all applications of machine learning (Deo, 2015).

1.2 Applications of Machine Learning in the Medical Field

From the advent of artificial intelligence in computers, the medical field has been one of many industries to benefit greatly from machine learning. A common practice in the medical field is giving diagnoses of conditions to patients, as well as predicting whether or not patients may be susceptible to certain ailments (Gharagyozyan, 2019). By using the prediction and classification abilities of trained machine learning models, data scientists and doctors are able to produce more accurate diagnoses (Gharagyozyan, 2019). Where applicable, computer vision models can be used on imaging data such as x-rays to detect tumors and other unseen abnormalities (Gharagyozyan, 2019).

1.2.1 Predictive Diagnosis

Machine learning can be used to detect patterns within the health records of patients who have been diagnosed with a specific ailment (Gharagyozyan, 2019). For example, if patients are tested for a type of cancer, those confirmed to have said cancer may have similar segments of data compared to individuals who do not have cancer. This form of advanced pattern recognition can be implemented with either complex or shallow neural networks (Gharagyozyan, 2019). As a classification problem, machine learning models would utilize supervised learning to measure the accuracy of a model's produced classifications (Gharagyozyan, 2019). An example of data metrics a machine learning model might be trained on is depicted in **Figure 4** below.

Figure 4

	Attribute	Domain
1	Sample code number	ID number
2	Clump Thickness	1 - 10
3	Uniformity of Cell Shape	1 - 10
4	Marginal Adhesion	1 - 10
5	Single Epithelial Cell Size	1 - 10
6	Single Epithelial Cell Size	1 - 10
7	Bare Nuclei	1 - 10
8	Bland Chromatin	1 - 10
9	Normal Nucleoli	1 - 10
10	Mitoses	1 - 10
11	Class	2 for benign, 4 for malignant

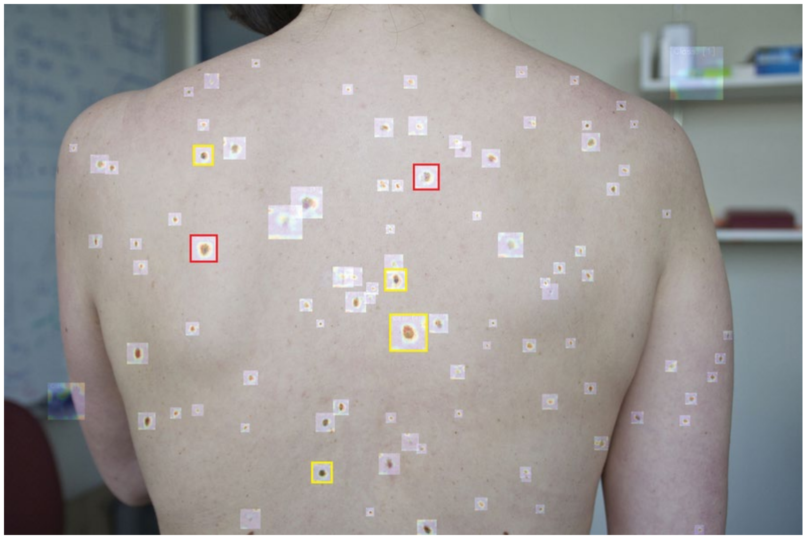
Characteristics of Benign and Malignant Tumors. Reprinted from Macadamian, by H. Gharagyozyan, 2019. https://www.macadamian.com/learn/a-practical-application-of-machine-learning-in-medicine/

As in most cases of machine learning, it is critical to find the optimal algorithms and modeling techniques necessary to produce the most accurate diagnosis predictions. Depending on the complexity of a model, larger amounts of data may be required to prevent overfitting (IBM Cloud Education, 2021). Overfitting occurs when a model, usually provided with insufficient data, fits exactly against its training data (IBM Cloud Education, 2021). Models suffering from overfitting can be identified by their unusual accuracy numbers, such as 50.000% (IBM Cloud Education, 2021). When working with smaller data sets, it may be better to use less complex models to combat the overfitting of training data. On larger data sets, where overfitting is not likely to become an issue, other methods can be used to produce greater model accuracy, such as data augmentation (IBM Cloud Education, 2021).

1.2.2 Computer Vision on Imaging Data

Medical imaging technology is a revolutionary technology that has been used for decades to identify problems within the body that may not appear externally (Vemuri, 2021). While some issues, such as broken bones, can be easily distinguished with the naked eye, other complex issues like cancerous melanoma can be rather difficult to discern (Vemuri, 2021). By using computer vision, medical professionals are able to make conclusions otherwise unavailable using pure speculation. Computer vision is a type of supervised machine learning that makes one or many classifications based on image data (Deo, 2015). Like all supervised learning models, computer vision models are trained with data sets unique to a specific task, like identifying cancerous melanoma (Vemuri, 2021). A unique attribute of computer vision is being able to see the regions in which classifications are made. In **Figure 5** below, boundary boxes are used to show which regions of the skin have been classified as likely to have cancerous melanoma.

Figure 5

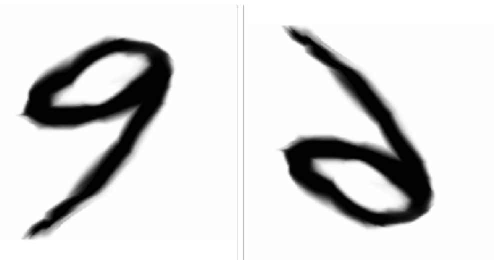


Detecting Melanoma Using Neural Networks. Reprinted from MIT News by M. Lewis, 2021.

https://news.mit.edu/2021/artificial-intelligence-tool-can-help-detect-melanoma-0402

If presented with an image of a horse upside-down, a human would likely be able to classify the image as a horse with ease. For computer vision models to emulate this ability, data augmentation can be used on a model's training data set (Zaworksi, 2020). Data augmentation can be any transformation performed on an image, such as cropping, stretching, recoloring, or rotating (Zaworksi, 2020). Data augmentation not only allows for a wider understanding of images but also greatly increases the amount of training data available for the model to learn (Zaworksi, 2020). However, as Radoslaw Zaworksi (2020) states, "there are situations when data augmentation should be used with caution, such as in medical applications because here augmentation will change their labels". An example of how data augmentation could potentially alter the meaning of data is provided below in **Figure 6.** By changing the orientation of the digit nine, it begins to more closely resemble the number six.

Figure 6



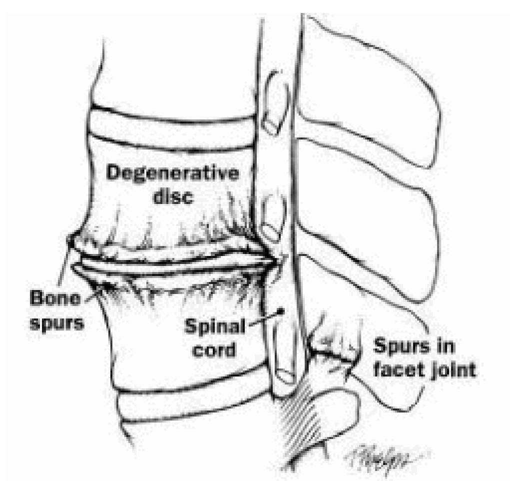
Data Augmentation Changing Data Meaning. Reprinted from Medium, by R. Zaworski, 2020. https://medium.com/snowdog-labs/data-augmentation-techniques-and-pitfalls-of-small-datasets-e5a657fc404f

1.3 What Is Degenerative Disc Disease?

The spine is one of the most important structures of the human body, allowing for every function of motion. Also referred to as the vertebral column, the spine is made up of thirty-three column-like bones called vertebrae (Columbia University Irving Medical Center, 2021). The vertebrae are grouped into three regions, cervical (near the neck), thoracic (near the upper back), and lumbar (near the lower back) (Columbia University Irving Medical Center, 2021). Each pair of vertebrae is separated by a cartilaginous disc, which is used to cushion the vertebrae from coming in contact with one another (Columbia University Irving Medical Center, 2021). From age progression and through extensive strain, the discs cushioning the vertebrae will begin to change (Columbia University Irving Medical Center, 2021). The changes in which the discs undergo will sometimes cause them to bulge, herniate, or thin out (Columbia University Irving Medical Center, 2021).

According to Dr. Brian McHugh (2017), "Degenerative disc disease is one of the most common causes of low back and neck pain". Degenerative disc disease refers to symptoms of back or neck pain caused by prolonged degradation of a spinal disc (McHugh, 2017). It should be noted that "degenerative" within the name of the disease does not mean the disc will continue to degenerate, merely that the disc has degenerated already. A degenerated disc will typically cause chronic, low-intensity pain with occasional episodes of sharp pain (McHugh, 2017). The pain experienced by those affected by degenerative disc disease has been described as numb, hot, and/or shooting (McHugh, 2017). This pain is more commonly experienced in the vertebrae near the neck and upper back, as these areas of the spine are subject to most stress (McHugh, 2017). **Figure 7** depicts what a degenerated disc might look like in a patient.

Figure 7



Depiction of Degenerated Disc. Reprinted from Johns Hopkins, by Johns Hopkins, 2021. https://www.hopkinsmedicine.org/health/conditions-and-diseases/degenerative-disc-disease. Copyright 2021 by The Johns Hopkins University, The Johns Hopkins Hospital, and Johns Hopkins Health Systems.

Degenerative disc disease is fairly simple to diagnose using medical imaging technology. Patients with mild cases of degenerative disc disease can usually be diagnosed with an x-ray of the spine (Columbia University Irving Medical Center, 2021). Though, in patients with more severe cases, an MRI or CT scan may be used to determine the exact extent of damage a disc has experienced (Columbia University Irving Medical Center, 2021). Imaging diagnosis is typically recommended by spinal treatment professionals, such as chiropractors (Grassi, 2019). Chiropractors will often weigh past and present patient records along with symptoms currently experienced to determine the probability of degenerative disc disease (Grassi, 2019). Besides records, a chiropractor might observe the mechanics of the spine, such as gait, posture, and range of motion to diagnose degenerative disc disease (Grassi, 2019). If it is believed by a chiropractor that a patient has degenerative disc disease, a referral may be given to perform a proper imaging diagnosis (Grassi, 2019).

Degenerative disc disease affects individuals of a broad spectrum but is most commonly seen in older patients (Stanford Health Care, n.d.). Many factors can play a part in whether or not a disc begins to degenerate, but a degenerative disc can be summarized as a function of strain over time (Stanford Health Care, n.d.). Individuals who may be overweight or working in labor-intensive fields would exert more strain on their spine and therefore be at risk of developing degenerative disc disease at an earlier age, whereas those who do not perform heavy labor and are not overweight might experience degenerative disc later in life (Stanford Health Care, n.d.). Additional risk factors for degenerative disc disease include spinal disorders, traumatic injury, calcium deficiencies, early menopause, and bone disease (Stanford Health Care, n.d.).

Data Collection

2.1 Retrieving Medical Data

The efficacy of collecting medical data depends entirely on the organization of the data provider. Retrieving data can be, in some instances, as simple as querying a database to receive a CSV (comma separated values) file. Though, some industries and smaller providers of data may have entirely paper records that need to be manually entered or scanned. If the latter option is the case, data analysts can utilize a variety of tools to minimize manual data entry. The use of these tools as well as the ability to view sensitive medical data does not come without stipulations. When handling sensitive medical documents, it is important to abide by federal and state legislation. This may mean redacting any documents or data stored of information that could be used to identify an individual.

To train the machine learning models necessary for this thesis, a chiropractic office agreed to provide patient medical data for use in data analysis. The chiropractic office was able to query a list of patient ID numbers which were affected with degenerative disc disease. However, all patient metrics and health data were kept as physical records, meaning that data entry would be required to perform analysis. A non-disclosure and HIPAA compliance agreement was signed to allow for data collection. In entering patient health information, no form of identifiable data was stored, in compliance with HIPAA guidelines.

2.2 NDA and HIPAA

In many workplaces that deal with sensitive information or trade secrets, an employee or contractor is made to sign a non-disclosure agreement. A non-disclosure agreement is a legally binding contract that prevents the signee from sharing any sensitive information pertaining to their business (Twin, 2021). In more competitive fields, a non-disclosure agreement can be used to prevent competitors from gaining trade secrets. In medical fields, a non-disclosure agreement typically aligns with HIPAA legislation in the United States (Twin, 2021). HIPAA, or the Health Insurance Portability and Accountability Act of 1996, prevents the disclosure of any patient information which could be used to identify an individual (OCR, 2013). Identifiable information can be an individual's name, contact information, social security number, location data, etc. Failure to comply with HIPAA legislation can result in fines ranging anywhere from $100 to $50,000 per violation (OCR, 2013).

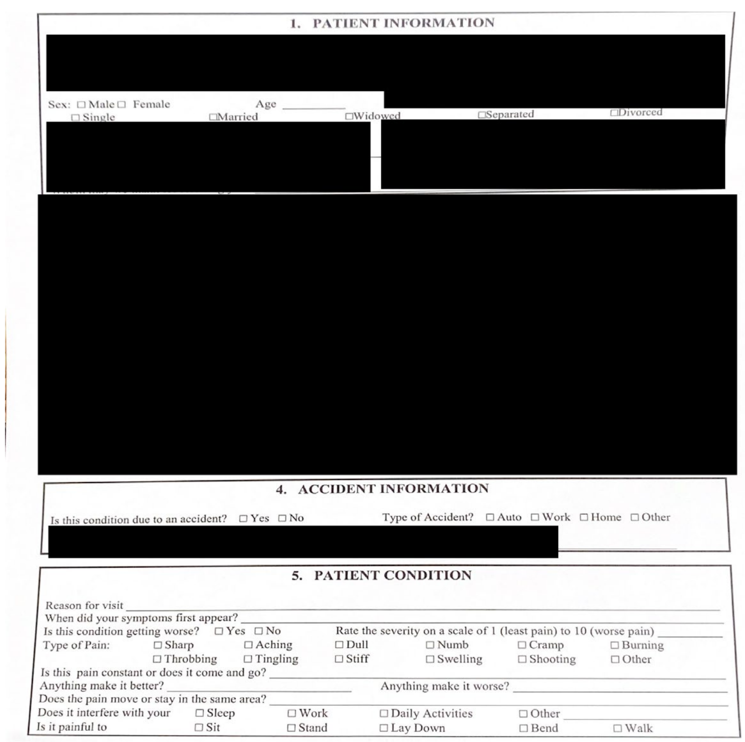
2.3 Data Entry Attempts

In collecting the data used to create the machine learning models in this thesis, precautions were taken to prohibit the entry of any identifiable information. Two modes of collecting data were theorized in order to efficiently enter all information. An efficient, but more complex, method of data entry, would involve computer vision machine learning models. A more obvious method of data entry would be to create a form that would export all entered data into a readable format. Both formats and their respective successes and shortcomings will be detailed in the sections below.

2.3.1 Computer Vision

Many streamlined file scanning applications have been created to allow for physical documents to be converted to digital PDFs. An attempt was made to create a program implementing Adobe Scan API which would allow physical health records to be made into electronic records as PDFs. Once converted, the PDF would be corrected to a standardized orientation. Because all scanned health records follow the same page structure, an overlaying image would be applied on top of the PDF to essentially redact all sections of identifiable patient data. By applying computer vision neural networks, the final image of a patient's health record would be read, with the resulting data being inserted into a CSV file. **Figure 8** depicts the resulting product of a patient's health record scanned into the proposed computer vision model.

Figure 8



1. PATIENT INFORMATION

[redacted]

Sex: ☐ Male ☐ Female Age ________ [redacted]

☐ Single ☐Married ☐Widowed ☐Separated ☐Divorced

[redacted]

4. ACCIDENT INFORMATION

Is this condition due to an accident? ☐ Yes ☐ No Type of Accident? ☐ Auto ☐ Work ☐ Home ☐ Other

[redacted]

5. PATIENT CONDITION

Reason for visit ________

When did your symptoms first appear? ________

Is this condition getting worse? ☐ Yes ☐ No Rate the severity on a scale of 1 (least pain) to 10 (worse pain) ________

Type of Pain: ☐ Sharp ☐ Aching ☐ Dull ☐ Numb ☐ Cramp ☐ Burning

☐ Throbbing ☐ Tingling ☐ Stiff ☐ Swelling ☐ Shooting ☐ Other

Is this pain constant or does it come and go? ________

Anything make it better? ________ Anything make it worse? ________

Does the pain move or stay in the same area? ________

Does it interfere with your ☐ Sleep ☐ Work ☐ Daily Activities ☐ Other ________

Is it painful to ☐ Sit ☐ Stand ☐ Lay Down ☐ Bend ☐ Walk

Example Output of Computer Vision Model.

Upon reviewing the physical patient records, a computer vision model was deemed unfit to handle the task of entering the data in a HIPAA compliant manner. Due to a majority of the patient health record being entered by the patients themselves, many notation errors were found in the documents. The errors would be easily discernible to a human reviewing the record, but a computer vision model may very easily confuse the information, making incorrect classifications of data. A common error occurred when patients checked the "Rheumatoid Arthritis" box on the patient health record, only to scratch out the word "Rheumatoid". This would occur despite "Arthritis" being an additional box to check if a patient suffered from only arthritis. Similarly, patients might leave certain lines blank, filling in the information meant for those lines in the margins of the form. Ultimately, it was decided that the post-processing necessary to remedy erroneous patient records would take far longer than manual data entry.

2.3.2 Localized Data Entry Tool

After determining computer vision to no longer be a viable option, consideration of data entry methods began. The final destination of any data gathered, whether through computer vision or manual entry, would be in a CSV file. Rather than enter raw data into the CSV file by hand, a tool was created for data entry using HTML and Javascript. The tool mimicked the same layout seen in the physical patient records, though all fields requesting identifiable information were not included. Liberties were taken to trim the amount of unnecessary data given to a machine learning model. For instance, all "Yes or No" questions with open-ended response boxes were converted into binary choice radio buttons. Upon completion of a patient health record, the tool will append all entered data to a master CSV file.

Despite the tool being created with HTML and Javascript, the tool is not currently web-hosted. There are currently no plans to web-host the data entry tool, as any further work should focus on larger sets of data that could only feasibly be stored in databases. Approximately 160 patient records were entered manually using the localized data entry tool. This process took roughly 20 hours, not including retrieval and organization of the physical patient health records. With approximately 160 patient records in a single CSV file, all data required was ready to be implemented into machine learning models.

Machine Learning Models

3.1 Tabular Train - Fast AI

FastAI is an incredibly useful set of Python libraries used for creating machine learning models (Howard, 2018). Due to the tabular nature of the CSV file containing sample patient health information, it was determined that the Tabular Training library from FastAI may be most efficient for this thesis. Tabular Train from FastAI uses a library named Pandas to read tabular data from which machine learning models are created (Howard, 2018). The Tabular section of FastAI's libraries greatly assists in converting text data to numerical values. By converting text values to numerical values, the data becomes more portable to machine learning models in which text training is not an option.

3.1.1 Tabular Training Model

Tabular Training from FastAI utilizes many different classes from their extensive library (Howard, 2018). While the Tabular class is used to pre-process tabular data for training, the training function used, Tabular Train, is an object of the Learner class (Howard, 2018). The Learner class is used for supervised learning models where a training set contains true target values, "Y", and false target values, "X" (Howard, 2018). As is done commonly in supervised learning models, the Learner class allows for training data to be split at a designated margin for validating prediction accuracy (Howard, 2018). For the Learner class to train a model, it must be given a DataLoaders object and batch size as parameters (Howard, 2018).

When reading tabular data using the Pandas library, all data is saved as a DataFrame object (Howard, 2018). A DataFrame can be summarized as a table of data where labels are given to each column, and each row is assigned a number. Once a DataFrame is modified according to need, the DataFrame can be converted into a DataLoaders object through the "dataloaders()" method (Howard, 2018). DataLoaders contains multiple DataLoader objects which can be simplified as a row of data bundled with other relevant information (Howard, 2018). **Figure 9** displays a visual representation of a DataLoaders object.

Figure 9

	Sex	Accident	AccType	Concern	WhenSymptoms	Bettering	IsSharp	isAching	isDull	IsNumb	isCramp	isBu
0	Female	No	#na#	pain in shoulder blades	August	#na#	False	True	False	False	False	
1	Male	No	#na#	lower back pain	2 days prior	#na#	True	False	False	False	False	
2	Female	No	#na#	shoulder and back pain	#na#	Yes	False	True	False	False	False	
3	Female	No	#na#	pain in the shoulder	a few weeks prior	Yes	False	False	False	False	False	
4	Male	No	#na#	neck pain	1 month prior	#na#	False	True	False	False	False	
5	Female	No	#na#	muscle therapy	3 months prior	Yes	False	False	False	False	False	
6	Male	No	#na#	posture	10 years prior	No	False	False	True	False	False	
7	Female	No	#na#	tennis elbow	4 months prior	#na#	False	True	True	True	False	
8	Female	No	#na#	back pain	#na#	#na#	False	False	False	False	False	
9	Male	No	#na#	back pain	yesterday	No	True	False	False	False	False	

Example of DataLoaders Object.

While not explicitly stated in the documentation, it is believed that all training functions implementing the Learner class utilize regression algorithms (Howard, 2018). Multiple regression focuses on linear and non-linear correlations between inputs (Hayes, 2021). For the purpose of this thesis, the regression algorithm being used is multiple regression. Simple linear regression is represented by the formula seen in **Figure 10**, where the dependent variable "y" is determined by a weighted explanatory variable "x" (Hayes, 2021). The dependent variable "y" could also be considered a desired output, while each explanatory variable "x" represents an input.

Figure 10

$$y(x) = b_0 + b_1 x_1$$

Formula for Linear Regression

The multiple regression formula differs slightly, as a single dependent variable is compared against a multitude of explanatory variables (Hayes, 2021). **Figure 11** shows the modified formula for multiple regression.

Figure 11

$$y(x) = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_n x_n$$

Formula for Multiple Regression.

When training a regression model, the weighted coefficients in front of each explanatory variable are initialized based on correlation to the dependent variable (Hayes, 2021). Referring back to the DataLoaders visual in **Figure 9**, the explanatory variables are the data measured in each column of the DataLoaders object. For example, "$b_1 x_1$" could be the variable "Sex" from the

DataLoaders object in **Figure 9**, the coefficient "b_1" developing a correlation to the dependent variable, "Degenerative Disc".

3.1.2 Tabular Train Methodology

In order to use the FastAI libraries, all elements in the Tabular library were imported to a Jupyter Notebook file. Along with FastAI, the Pandas library was implemented to read the CSV file of sample patient health records. Using the “read_csv()” method from Pandas, the patient data was converted to a DataFrame object within the Jupyter Notebook. If an attempt was made to convert the DataFrame to a DataLoaders object in its current condition, many errors would occur. While the Tabular Train method from FastAI can be trained on text data, the raw patient data had many missing values. These missing values would be assigned as “null”, which in programming functions can lead to errors if null values are not expected. To avoid any possible errors, and to convert all categorical and text values into numerical data, the Tabular class from FastAI was used. The “TabularPandas()” method from FastAI takes a DataFrame and its specified categorical and continuous variables as parameters, converting all values to numerical values (Howard, 2018). Optionally, the “TabularPandas()” method can receive an additional “split” parameter to divide the DataFrame into training and validation sets (Howard, 2018).

After being modified, the DataFrame was converted to a DataLoaders object using the “dataloaders()” method. The “dataloaders()” method allows a batch size parameter to be passed, to which multiple batch sizes were attempted. With approximately 160 rows of data, the batch size used was 50. With the DataLoaders object created, the patient data was prepared to be used in the creation of a Tabular Train model. Both the DataLoaders object and a validation metric were passed as parameters to the “tabular_learner()” function. When the function was run, the training time was notably fast, though this could be due to the rather small training data size.

Once trained, a model named "learn" was created and subsequently prepared for validation testing.

Multiple split ratios were attempted in separating the training and validation sets of patient data. Ultimately, it was decided to use split ratios of 80/20 and 70/30 for training and validation sets. With the size of the total patient data being approximately 160 entries, the size of each validation set would be roughly 32 and 48 respectively. These sizes are important, as they indicate the maximum amount of testing epochs, or loops, performed on each validation set. In order to validate trained models, FastAI utilizes a Schedulers class (Howard, 2018). Within the Schedulers class is a function "fit_one_cycle()", which takes in a number of epochs as a parameter, and operates according to a policy proposed by Leslie N. Smith et al (Smith & Topin, 2018). The policy proposes a scheduler using a cosine annealing learning rate schedule where the learning rate is reset on each epoch (Smith & Topin, 2018). The mathematical model for the scheduler can be seen in **Figure 12**, where n_{min}^{i} and n_{max}^{i} are ranges for the learning rate and T_{cur} is the number of epochs performed (Loshchilov & Hutter, 2017).

Figure 12

$$\eta_t = \eta_{min}^{i} + \frac{1}{2}\left(\eta_{max}^{i} - \eta_{min}^{i}\right)\left(1 + \cos\left(\frac{T_{cur}}{T_i}\pi\right)\right)$$

Cosine Annealing Learning Rate. Reprinted from Papers With Code, by Loshchilov, et al, 2017. https://paperswithcode.com/method/cosine-annealing.

3.1.3 Tabular Train Results and Takeaways

In using the "fit_one_cycle()" method for validation, 15 epochs were used, resulting in the data seen in **Figure 13**.

Figure 13

```
# Testing
learn.fit_one_cycle(15)
```

epoch	train_loss	valid_loss	accuracy	time
0	0.617184	0.688827	0.677419	00:00
1	0.600651	0.688343	0.774194	00:00
2	0.574610	0.688785	0.677419	00:00
3	0.545882	0.689961	0.709677	00:00
4	0.515628	0.690853	0.741935	00:00
5	0.488816	0.690604	0.741935	00:00
6	0.465598	0.689001	0.645161	00:00
7	0.439373	0.685613	0.677419	00:00
8	0.414736	0.684625	0.677419	00:00
9	0.398137	0.685556	0.645161	00:00
10	0.373761	0.683971	0.677419	00:00
11	0.353848	0.682495	0.677419	00:00
12	0.335440	0.679719	0.677419	00:00
13	0.325686	0.680308	0.580645	00:00
14	0.309461	0.676818	0.612903	00:00

Tabular Train Results.

A notable occurrence is the lowering of the training loss with each successive epoch. This has little effect on the model's overall accuracy, with accuracy values fluctuating with each successive epoch. In this instance of training, the highest achieved accuracy was 0.774194, the lowest being .580645. When measuring recall and precision metrics, values varied greatly. As the data is randomized with each run of the training function, different training data would result in unique precision and recall values. Two separate instances allowed for a uniform recall of 1.0, with every other instance varying. Precision scores were typically very low, with occasional uniform distribution of 0.0. **Figure 14** contains a table of all average measured metrics from the Tabular Train model.

Figure 14

Max Acc	Min Acc	Avg Acc	Mdn Prec	Avg Prec	Mdn Rec	Avg Rec
.774194	.580645	.6344	.322581	.3379	.80	.880

Tabular Train Metrics.

From the data presented, it can be presumed the model is prone to overfitting. The oddly uniform distributions along with instances of precisely .500000 accuracy show the validation set is at times merely attempting to identify the data it is trained on.

3.2 Random Forests

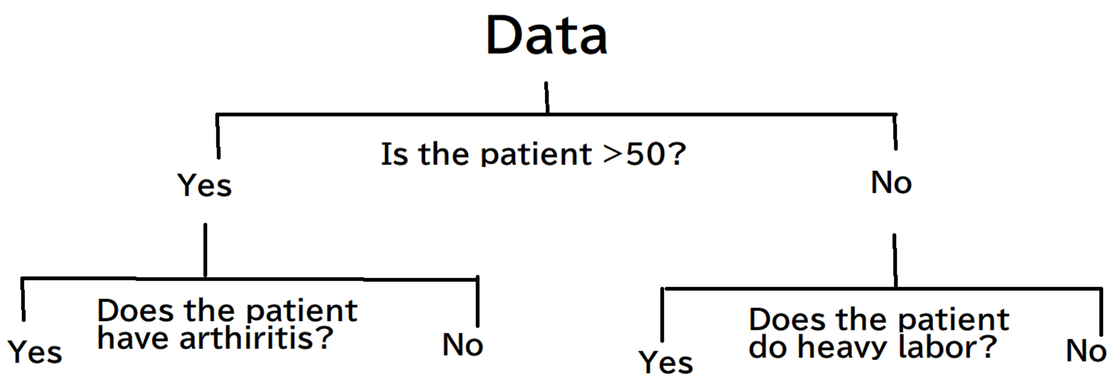
The Python library SciKit-Learn (abbreviated to SKLearn) contains many useful machine learning models, one being the Random Forest Classifier (Pedregosa et al., 2011). FastAI is incredibly portable to SKLearn machine learning functions, as FastAI includes many SKLearn base functions in their libraries (Pedregosa et al., 2011). Because of this, the modified data used in the Tabular Train model was able to be used in training the Random Forest Classifier model.

3.2.1 Random Forests

The random forest classifier is used in supervised learning models to solve classification problems (Yiu, 2021). A “forest” in the context of random forests refers to a group of decision tree classifiers (Yiu, 2021). Decision trees are essentially a series of yes or no questions that increase in levels of specificity as the tree is traversed (Yiu, 2021). Upon reaching the end of a tree, a classification is stored at its destination, which can be reached by following the same sequence it is encoded by (Yiu, 2021). The random forests classifier uses a collection of decision trees, with each tree having its own method of reaching a classification prediction (Yiu, 2021). The most precise tree will be used as the decision tree for the random forest classifier (Yiu, 2021).

Each decision tree in a random forest classifier is careful not to be too similar to other trees (Yiu, 2021). This is achievable through a process called feature randomness (Yiu, 2021). Feature randomness separates the features, or variables, that a tree can use as decision branches at each level (Yiu, 2021). While one tree may have access to, for example, variables 1 and 3 in deciding its next branch, another tree may only have access to variables 2 and 4. This technique ensures that a diverse group of trees is created in a random forest. An example of a decision tree subsection that could be within a random forest is displayed in **Figure 15.**

Figure 15



Example of Decision Tree.

3.2.2 Random Forests Methodology

Fortunately, due to the portable nature of the FastAI library, the data pre-processed for use in Tabular Train was able to be used almost entirely in the random forest classifier model from SKLearn. All pre-processed data was represented as a DataFrame object, which conveniently is the exact data type necessary as a parameter for the random forest classifier method. Due to the random forests classifier model needing different parameters, the DataFrame object was separated into four separate sets. The data to be contained within each set was predetermined by methods of the FastAI Tabular class. The four separate sets of data contained the differentiated positive and negative classifications of both the training and validation set.

With the data parsed in a manner readable to the SKLearn random forests classifier model, training and validation of the model were completed in merely three lines of code. First, the “RandomForestClassifier()” method was used to instantiate the classifier being used. Once instantiated, a generic SKLearn child method, “fit()”, was used to train the classifier. The “fit()” function merely trains the classifier using the model with which it was instantiated; in this case, the model was the random forests classifier. Both the positive and negative classification training sets were passed as parameters to the "fit()" function, allowing for the random forests classifier to be created. To perform validation of the newly trained model, the generic child function "predict()" was used, with the negative classification validation set passed as a parameter. The result was stored in a variable named "predicted", to which graphing operations and metric measurements could be performed.

3.2.3 Random Forests Results and Takeaways

While FastAi remains an incredibly powerful library, it was rather difficult learning how to retrieve multiple different metrics at once when attempting to measure the accuracy of a model. With the SKLearn library, measurements were fairly straightforward, mostly being used as separate child functions, rather than parameters of a validation function. The result of measuring the accuracy on different metrics can be seen in **Figure 16.**

Figure 16

```
Accuracy Score : 0.8064516129032258
Report :
              precision    recall  f1-score   support

           0       0.80      0.95      0.87        21
           1       0.83      0.50      0.62        10

    accuracy                           0.81        31
   macro avg       0.82      0.73      0.75        31
weighted avg       0.81      0.81      0.79        31
```

Random Forests Classifier Results.

From the average precision and recall scores seen in Figure 16, it appears the model was able to uncover enough correlation between the data to make reasonable predictions. While measured metrics did not approach the desired result of 99%, an average accuracy of 80% is certainly more accurate than a random guess. From the data measured, it is not entirely evident if overfitting played a role in the resulting averages, but it should not be ruled out as a possibility.

3.3 Data Modeling

Typically, one of the best ways to visualize data trends is through graphing. However, given that there were roughly 116 input variables to these models, graphing attempts were incredibly cluttered at times, offering very little comprehension of correlations. To circumvent the cluttered data of a graph while still gaining insight as to correlations learned by the model, the library ELI5 (Explain It Like I'm 5) was used. ELI5 is a library in Python used to visually represent the information within machine learning models (Jha, 2021). ELI5 is only compatible with the random forests classifier created from SKLearn. Due to the random forest classifier having much greater average accuracy, solely visualizing its model should serve as sufficient. **Figure 17** shows the output of the "show_weights()" function from ELI5, where the top twenty highest weighted input variables have been returned.

Figure 17

Weight	Feature
0.1276 ± 0.2307	Age
0.0635 ± 0.1214	Concern
0.0396 ± 0.1069	Surgeries
0.0349 ± 0.0794	WhenSymptoms
0.0281 ± 0.0780	BettPain
0.0256 ± 0.0632	Painscale
0.0254 ± 0.0819	hasWhoopC
0.0245 ± 0.0630	Worsepain
0.0227 ± 0.0763	TkChiro
0.0191 ± 0.0772	hasHernia
0.0179 ± 0.0518	Exercise
0.0161 ± 0.0634	hasEpilepsy
0.0160 ± 0.0622	hasTumors
0.0155 ± 0.0542	Sex
0.0129 ± 0.0553	pnLay
0.0125 ± 0.0461	intWork
0.0125 ± 0.0543	BonInj
0.0123 ± 0.0510	IsSharp
0.0122 ± 0.0611	hasUlcers
0.0121 ± 0.0432	Caffeine

Random Forests Classifier Weight Visualization.

The weights displayed for each input variable in **Figure 17** show what the random forests classifier has learned as important contributions towards classification. The strongest correlated input variable as determined by the model was a patient's age. Other highly weighted input variables were the patient’s primary concern, previous surgeries, and the first occurrence of an experienced symptom. It should be noted that the weights shown do not specify the value of any input variable. For example, the input variable named “TkChiro” refers to a patient previously receiving chiropractic care. While it is weighted as one of the top twenty variables, the value could be “yes” or “no”, positively or negatively correlating the values to the weight. To understand what values might contribute to the weights of input variables, a true positive classification was analyzed. **Figure 18** displays the output of the "show_prediction()" function from ELI5, where the values contributing towards a classification are ranked by learned weight.

Figure 18

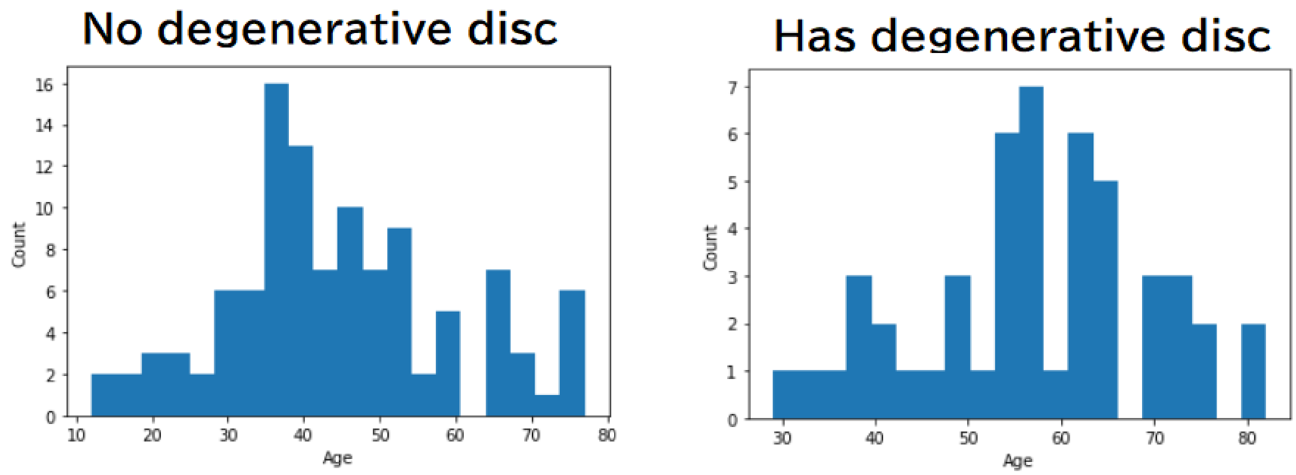
y=1 (probability **0.730**) top features

Contribution?	Feature	Value
+0.307	<BIAS>	1.000
+0.105	Age	65.000
+0.048	WhenSymptoms	0.000
+0.035	Sex	3.000
+0.035	hasTumors	2.000
+0.033	hasEmphysema	2.000
+0.032	Concern	60.000
+0.028	TK_PT	2.000
+0.023	AccType	1.000
+0.018	intDaily	2.000
+0.016	isThrobbing	2.000

Random Forests Classifier Prediction Weight Visualization.

The first value displayed in **Figure 18** is the bias value or the expected average score output by the model (Jha, 2021). In direct correlation with the general weights seen in the model, the highest weighted value in predicting a degenerative disc diagnosis was the patient's age. A visible difference between the model's weight of all input variables and this particular prediction is the absence of the concern and surgery variables in the top ten contributing factors. This implies the patient of whom the prediction was being made did not have a surgery or concern value that corresponded with what the model had learned to recognize. To better visualize at least two of the input variables learned by the model to be contributing factors, patients' age and surgery histories were graphed respectively. **Figure 19** shows histograms counting the age distribution of patients with and without degenerative disc disease.

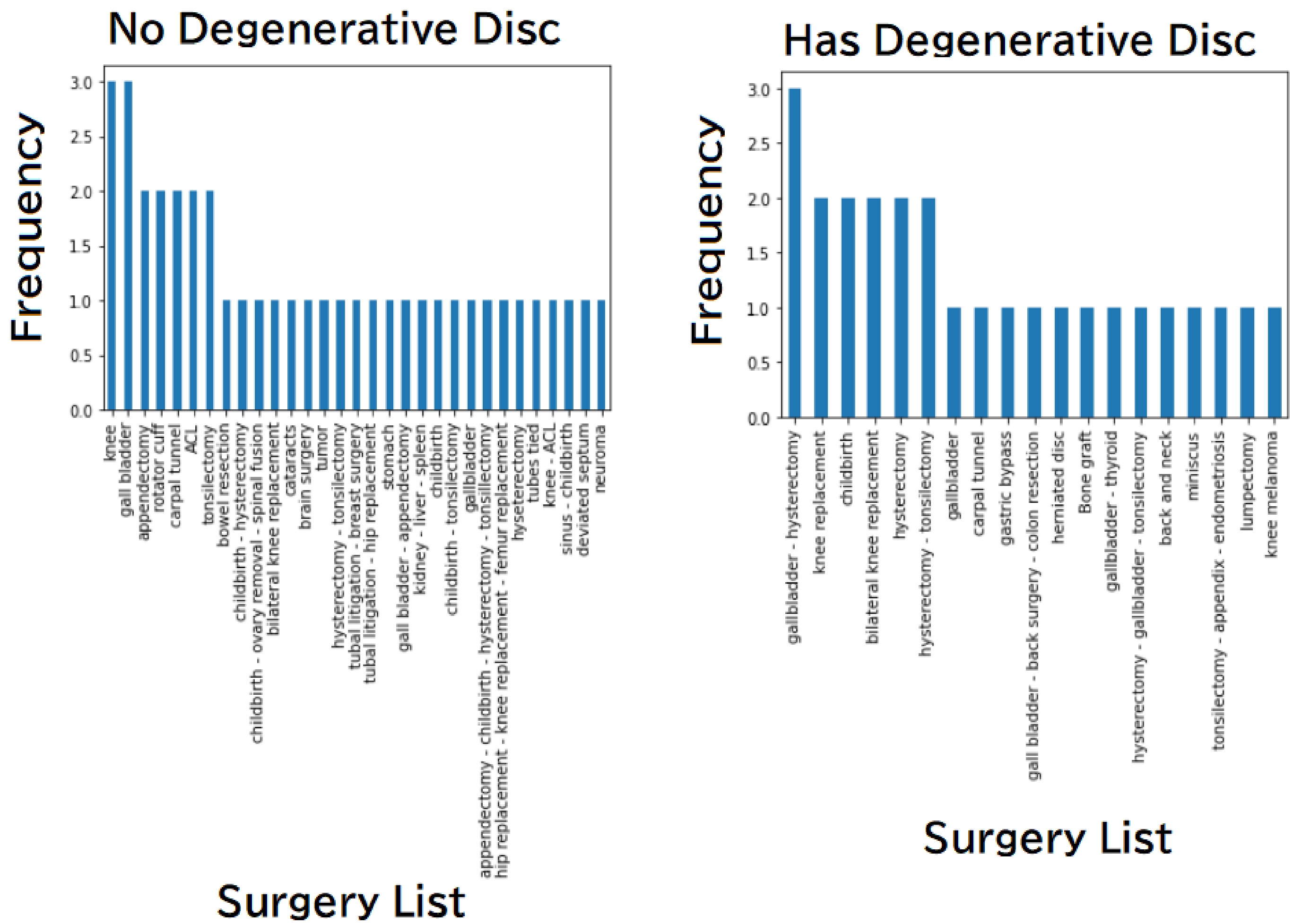
Figure 19



Distribution of Age in Patients With and Without Degenerative Disc Disease.

The histogram of patients who do not have degenerative disc disease shows a rather uniform distribution of age, with more older individuals than younger individuals. The histogram of patients who do have degenerative disc disease shows left-skewed data favoring older individuals. This data serves to indicate a correlation between older individuals and those diagnosed with degenerative disc disease. Given this observed correlation, it is good the model learned this to be of strong correlation. **Figure 20** shows bar charts displaying the frequency of surgeries in patients with and without degenerative disc disease.

Figure 20



Frequency of Surgeries in Patients With and Without Degenerative Disc Disease.

While the data is rather compact due to the nature of Python's graphing library, there are several surgeries shared amongst those who have been diagnosed with degenerative disc disease. Of course, there are many shared surgeries amongst those who do not have degenerative disc disorder as well, but the surgeries seen do not have much correlation respectively. Comparing the left and right bar chart of **Figure 20**, a surgery seen more frequently in patients with degenerative disc disease is a hysterectomy.

Conclusion

Two machine learning models, Tabular Train and random forests classifier, were trained on manually gathered patient health records. Despite the small data sample of roughly 160 entries, the random forests classifier was able to deliver rather promising results. The Tabular Train model had a top accuracy of 77% but showed clear signs of overfitting for subsequent training loops. For example, certain epochs would result in accuracies of precisely 50%, which should be incredibly unlikely to achieve. It is likely, based on a visual representation of weights, the random forests classifier also contained instances of overfitting. However, despite any possible overfitting, the random forests classifier was able to learn strong correlations between input variables and relevance to degenerative disc disease.

The highest weighted input variable of the random forests classifier model was a patient's age. When predicting degenerative disc disease on a certain patient record, the random forests classifier weighted said patient's age, 65, as a leading contributor to the classification. The model determining older individuals as more likely to suffer from degenerative disc disease correlates to the current medical knowledge of degenerative disc disease. While the model itself may not have the desired accuracy of 99%, there are strong enough correlations recognizable by the model such as to suggest the plausibility of predicting degenerative disc disease using machine learning. From this revelation, the aims of this thesis have been fulfilled.

Discussion

After completion of the thesis project and reaching a conclusion through machine learning models, an interview was held with Dr. Lori Wight DC. Dr. Wight, throughout the interview, shares her interpretations of the thesis conclusion, as well as ideas for application in the field.

5.1 Limitations

One of the most notable drawbacks of this thesis was the lack of patient data. Small sources of data in machine learning models often leads to overfitting, which can hurt the overall effectiveness of the model. The small data sample was partially the result of another limitation, physical patient health records. Manual entry of data is very time-consuming and is not favorable when research is dependent on the data to progress further. Upon entering the data, it became apparent each patient record contained data mostly filled by the patient. While the data being patient entered does not invalidate it, sanitization of the data upon entry was necessary to prevent semantics from confusing the model. For example, if a patient writes, “back pain” and another writes “pain in back”, the text is different, but the meaning can be summarized as “back pain”.

5.2 Speculation

Before the machine learning models were created and conclusions were drawn, hypotheses regarding data correlation were formed by reviewing the raw data. Specifically, in patients affected by degenerative disc disease, many shared values became more evident in subsequent graphing efforts. A major speculation, later backed by the random forests classifier and further medical research, was the association between age and degenerative disc disease. In an interview with Dr. Lori Wight, when asked what factors contribute to degenerative disc disease, Dr. Wight responded with, “Definitely age, but also things like trauma to the spine, back injuries” (Harding & Wight). In viewing the weights of the machine learning model, a correlation was seen between patients who had undergone hysterectomies and the development of degenerative disc disorder. When asked about her thoughts on this correlation, Dr. Wight commented, “A contributing factor in women with degenerative disc is menopause, so that certainly lines up with hysterectomies, as they bring on menopause” (Harding & Wight).

With a precision score of 80%, some diagnoses were misclassified by the model. When asked what confusion might occur when attempting to discern degenerative disc in patients, Dr. Wight replied, “there [will] probably be a gray area for middle-aged people who do blue-collar jobs. There [is] a chance they have degenerative disc, or maybe they just strained their back. Sometimes it can be hard to tell” (Harding & Wight). The machine learning model was trained on a rather small data set, though was able to make some correlations that bear weight to observed medical science. When asked if she believed a machine learning model could make distinctions similar to that of a chiropractor, Dr. Wight stated, “It could, though it would need some good data. [Chiropractors] measure things like posture, walking … so as long as those were noted I think it could, just as accurately” (Harding & Wight). When asked further if she thought machine learning models would be sufficient for diagnosis, Dr. Wight responded, “[Chiropractors] are typically pretty accurate; but even then, to one hundred percent diagnose degenerative disc you would have to have an x-ray taken” (Harding & Wight).

5.3 Future Work

From the correlations the random forest classifier was able to achieve on a small data sample, along with testimony from Dr. Lori Wight, it is possible machine learning models could greatly assist in the prediction of degenerative disc disease. To achieve models with higher accuracy, it may be beneficial to not only increase the sample size of the data but also increase the amount of data being recorded. With a large portion of good medical data, it could be possible to train machine learning models that are able to achieve the desired 99% classification accuracy. If the medical data used was patient entered, similar to the data used in this thesis, it could allow for interactive prediction models. For example, if a model has been trained with patient-entered data, a web application could be designed to allow prospective patients to receive a diagnosis prediction based on input. This could reduce potentially unnecessary examinations and allow for degenerative disc disease to be identified and treated at an earlier date.

List of References

Columbia University Irving Medical Center. (2021, July 21). *Degenerative disc disease*. Columbia Neurosurgery in New York City. Retrieved November 24, 2021, from https://www.neurosurgery.columbia.edu/patient-care/conditions/degenerative-disc-disease.

Delua, J. (2021, March 12). *Supervised vs. unsupervised learning: What's the difference?* IBM. Retrieved November 18, 2021, from https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning.

Deo, R. C. (2015, November 17). *Machine learning in medicine*. PMC Labs. Retrieved November 21, 2021, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5831252/.

Gharagyozyan, H. (2019, October 10). *A practical application of machine learning in medicine*. Macadamian. Retrieved November 18, 2021, from https://www.macadamian.com/learn/a-practical-application-of-machine-learning-in-medicine/.

Grassi, R. (2019, June 26). *Chiropractic care for degenerative disc disease*. Spine Universe. Retrieved November 21, 2021, from https://www.spineuniverse.com/conditions/degenerative-disc/chiropractic-care-degenerative-disc-disease.

Hayes, A. (2021, March 30). *Multiple linear regression (MLR) definition*. Investopedia. Retrieved November 24, 2021, from https://www.investopedia.com/terms/m/mlr.asp.

Howard, J. (2018). *fastai*. FastAI. Retrieved November 5, 2021, from https://github.com/fastai/fastai.

IBM Cloud Education. (2020, July 15). *What is machine learning?* IBM. Retrieved November 19, 2021, from https://www.ibm.com/cloud/learn/machine-learning.

IBM Cloud Education. (2021, March 3). *What is overfitting?* IBM. Retrieved November 21, 2021, from https://www.ibm.com/cloud/learn/overfitting.

Jha, A. (2021, July 19). *ML model interpretation tools - what, why, and how to interpret*. Neptune. Retrieved November 28, 2021, from https://neptune.ai/blog/ml-model-interpretation-tools.

Johns Hopkins. (n.d.). *Degenerative disc disease*. Johns Hopkins Medicine. Retrieved November 26, 2021, from https://www.hopkinsmedicine.org/health/conditions-and-diseases/degenerative-disc-disease.

Lewis, M. (2021, April 2). *An artificial intelligence tool that can help detect melanoma*. MIT News | Massachusetts Institute of Technology. Retrieved November 21, 2021, from https://news.mit.edu/2021/artificial-intelligence-tool-can-help-detect-melanoma-0402.

Loshchilov, I., & Hutter, F. (2017, May 3). *SGDR: Stochastic gradient descent with warm restarts*. arXiv.org. Retrieved November 30, 2021, from https://arxiv.org/abs/1608.03983v5.

McCarthy, J. (2004, November 24). *What Is Artificial Intelligence?* Formal - Stanford. Retrieved November 10, 2021, from https://borghese.di.unimi.it/Teaching/AdvancedIntelligentSystems/Old/IntelligentSystems_2008_2009/Old/IntelligentSystems_2005_2006/Documents/Symbolic/04_McCarthy_whatisai.pdf.

McHugh, B. (2017, December 13). *What is degenerative disc disease?* Spine-health. Retrieved November 18, 2021, from https://www.spine-health.com/conditions/degenerative-disc-disease/what-degenerative-disc-disease.

OCR, O. for C. R. (2013, July 26). *Summary of the HIPAA security rule*. HHS.gov. Retrieved November 26, 2021, from https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Santos, M. (2020, May 18). *Explaining precision vs. recall to everyone*. Medium. Retrieved November 26, 2021, from https://towardsdatascience.com/explaining-precision-vs-recall-to-everyone-295d4848edaf.

Shung, K. P. (2020, April 10). *Accuracy, precision, recall or F1?* Medium. Retrieved November 21, 2021, from https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9.

Smith, L. N., & Topin, N. (2018, May 17). *Super-convergence: Very fast training of neural networks using large learning rates*. arXiv.org. Retrieved November 30, 2021, from https://arxiv.org/abs/1708.07120v3.

Stanford Health Care. (n.d.). *Stanford Health Care - Risk factors*. Stanford Health Care (SHC) - Stanford Medical Center. Retrieved November 25, 2021, from https://stanfordhealthcare.org/medical-conditions/back-neck-and-spine/degenerative-disc-disease/risks.html.

TURING, A. M. (1950). I.—Computing Machinery and intelligence. *Mind*, *LIX*(236), 433–460. https://doi.org/10.1093/mind/lix.236.433

Twin, A. (2021, September 14). *Non-disclosure agreement (NDA)*. Investopedia. Retrieved November 26, 2021, from https://www.investopedia.com/terms/n/nda.asp.

Vemuri, V. (2021, June 4). *The role of Computer Vision in the medical field*. Medium. Retrieved November 16, 2021, from https://medium.com/techtalkers/the-role-of-computer-vision-in-the-medical-field-6ebe2ae8b332.

Yiu, T. (2021, September 29). *Understanding random forest*. Medium. Retrieved November 30, 2021, from https://towardsdatascience.com/understanding-random-forest-58381e0602d2.

Zaworski, R. (2020, February 14). *Data augmentation techniques and pitfalls of small datasets*. Medium. Retrieved November 23, 2021, from https://medium.com/snowdog-labs/data-augmentation-techniques-and-pitfalls-of-small-datasets-e5a657fc404f.

List of Figures