Machine Learning for Understanding Aging

Design Document

sdmay20-41
Dr. Julie Dickerson
Ian Simon - Chief Engineer
Jacob Laing - Chief Engineer
Nathan Carter - Test Engineer
Samantha Williams - Meeting Scribe
Scott Rose - Meeting Facilitator
Aria Sheets - Report Manager
sdmay20-41@iastate.edu
http://sdmay20-41.sd.ece.iastate.edu/

3 November 2019 / Version 2

Executive Summary

Development Standards & Practices Used

1. Our product must ensure the privacy of the people whose data we are using to use in the creation of our machine-learning program.
    a. All Personal Health Information (PHI) must be anonymized.
    b. Any PHI that is transmitted must be encrypted.
2. Our product must be accessible, and the information output must be easily understandable.
3. Our product must be fast to learn, and up to today's standards of machine learning.
4. Our product must output results that are accurate so that others can use the data we obtain through our program with reliability.

Summary of Requirements

Functional Requirements:

- Program accurately assesses patterns in data related to aging.
- User can continue to input data to increase the accuracy of the program.
- Program outputs aspects of the input data that affects the process or experience of aging.'
- Project completed by May 2020.

Non-Functional Requirements:

- Written in Python.
- Clear, well-documented code.
- Privacy of subjects included in test data is considered.
- Appropriate size of training data is used to properly train the program.
- The results of the running program are outputted in a user friendly format.

Applicable Courses from Iowa State University Curriculum

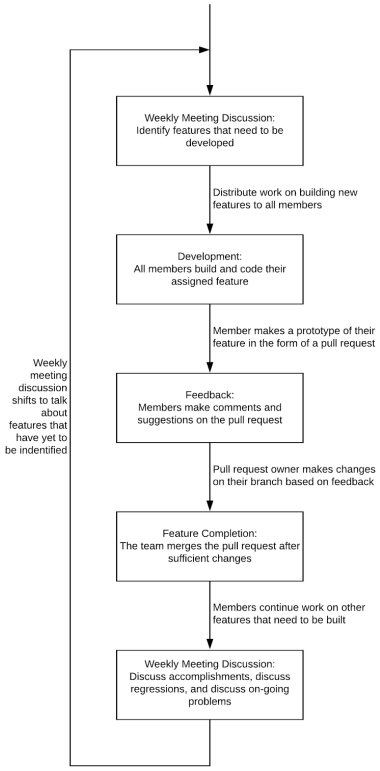
- COM S 227: Object-oriented Programming
- COM S 228: Introduction to Data Structures
- COM S 311: Introduction to the Design and Analysis of Algorithms
- COM S 474: Introduction to Machine Learning
- MATH 207: Matrices and Linear Algebra
- S E 309: Software Development Practices
- S E 329: Software Project Management
- S E 339: Software Architecture and Design

New Skills/Knowledge Acquired That Was Not Taught In Courses

- Neural network design.
- Cost functions used in neural networks.
- Analyzing large data sets.
- Managing privacy of data used programs for analyzing datasets.
- Regression analysis.

Table of Contents

List of figures/tables/symbols/definitions (This should be the similar to the project plan)



1. Introduction

1.1 Acknowledgement

Thank you to Dr. Julie Dickerson for providing guidance and structure for this endeavor. Thank you to Inter-University Consortium for Political and Social Research (ICPSR) for providing the large data sets that were used in the training and creation of the program. Thank you to Iowa State University for allowing us to use their hardware to run and test our program.

1.2 Problem and Project Statement

General problem statement: Human aging is a topic that has been studied throughout history. Scientists, doctors, sociologists, and the general public all want to know what characteristics indicate a decline in health and what actions can be taken to slow down this decline. Knowing this information can allow people to increase their life expectancy and overall quality of life.

General solution approach: Using health data collected by ICPSR during the *Midlife in the United States* (MIDUS) *Biomarker Projec*t, by the use of sensors and questionnaires, and research regarding machine-learning techniques, our product will find and return the various patterns found in the data. After training the program using large quantities of training data, it will be able to accept data and return information regarding actions in which the user can take that have proven effective for other patients with similar characteristics.

1.3 Operational Environment

Because our end-product requires mining a large set of given data to make the machine learn and output results, we recommend that our software is run on a high-end machine, preferably one with a high-end GPU. We will be developing our software expecting our users to be using a high-end machine. If they do not, the time for results to appear from our product will take longer. We will be using an Iowa State University Virtual Machine during the development of our product, and will strive to host a service on an Iowa State Virtual Machine for others to access our project.

1.4 Requirements

Functional Requirements:

- Program accurately assesses patterns in data related to aging.
- User can continue to input data to increase the accuracy of the program.
- Program outputs aspects of the input data that affects the process or experience of aging in a visual/graphical form.
- Project completed by May 2020.

Non-Functional Requirements:

- Written in Python.
- Clear, well-documented code.
- Privacy of subjects included in test data is considered.
- Appropriate size of training data is used to properly train the program.
- The results of the running program are outputted in a user friendly format.

1.5 Intended Users and Uses

The users of our end product will mostly consist of sociology and psychology scientists. They will be using our product to analyze data in hopes of finding patterns related to aging.

1.6 Assumptions and Limitations

Assumptions

- The program will be used by scientists and researchers.
- The program will be ran on high-end hardware.
- The program will produce results that correlate body movement to age.

Limitations

- The software will only be used by scientists and researchers.
- The budget is limited to hardware owned by the team members and hardware owned by Iowa State University.
- The data input into the program must fit a specific format.
- The program will only have English as the language.
- The program will rely heavily on GPU usage.
- The program needs to be completed by the beginning of May 2020.

1.7 Expected End Product and Deliverables

Our first major deliverable would to get the data parser and data formatter set up. These are necessary parts to be completed before we can expect out program to learn data. Even though this is our first deliverable, it is expected that we can work on the machine learning module by using dummy or mock data until the data parser and formatter is set up. The expected delivery date for this is late January 2020.

Our second major deliverable would to get our database set up, linking it to our data formatter module and our machine learning module. The database would read in data from the formatter module, storing the formatting data into the database to reduce duplicate formatting. The machine learning module will then read in the data, but no learning will be implemented yet. The database will also need to support the future implementation of storing the results after the machine learning module is setup. The expected delivery date for this is mid-late February 2020.

Our third major deliverable will be to create a proof of concept machine learning module. This module will use a small subset of our larger dataset. This subset will already have been studied by outside sources. Our goal with this proof of concept module will be to prove that our machine learning algorithms is correct. We will prove our algorithms correctness by comparing the results with ones already discovered. The expected delivery date for this is early-mid March. The proof of concept will include recreating a study that has been previously done on the MIDUS dataset. If we can recreate a study, we know that we have our data filtered appropriately and that our regression model works

Our fourth major deliverable is to create the machine learning module. This part is where all the machine learning takes place. This module reads in formatted data from the database and produces a result in the form of data from the learning process. This data is then sent and stored in

our database, and it will be sent to a data visualizer when it gets developed. The expected delivery date for this is late March 2020 to early April 2020.

Our last major deliverable is to create the result data parser and visualizer modules. These are the final pieces of our program, and will allow scientists/researchers to visualize the data in a friendly format, as well as show the pure data results of learning. The expected delivery date for this is late April 2020.

In the end, we will have created a program that can receive and analyze large datasets concerning aging and its effects. The program would use machine learning to possibly view this data in a different light than conventional methods. Our clients will be provided with our findings, as well as a framework for using the trained program on new sets of data. The hope will be that our program can provide new insights into how aging affects the body and mind, and perhaps lead to new advances in how we confront the universal challenge of getting old.

2. Specifications and Analysis

2.1 Proposed Design

We will be using TensorFlow in Python to create an unsupervised machine learning algorithm. The program will take in data from the MIDUS (Midlife in the United States) study conducted at the University of Michigan which already anonymized data to protect the privacy of individuals. The program will determine outcomes of aging based on the data input into the program. The program will run on a GPU cluster provided by Iowa State University. The data will be produced in a readable format so that it can be presented to scientists interested in our findings.

Currently, we haven't created code or tested anything that is part of the project so far, we have only researched the building blocks required to understand how machine learning functions before diving in. This includes research on concepts like neural networks and back-propagation. We have watched simulations and videos discussing machine learning and have inspected and compiled machine learning code that has been shown to work.

We have also completed trial runs of regression analyses with other data sets in the sklearn database. We did this in order to understand which type of regression will best fit the data we hope to use for our project. We decided that the lasso regression was the most effective.

2.2 Design Analysis

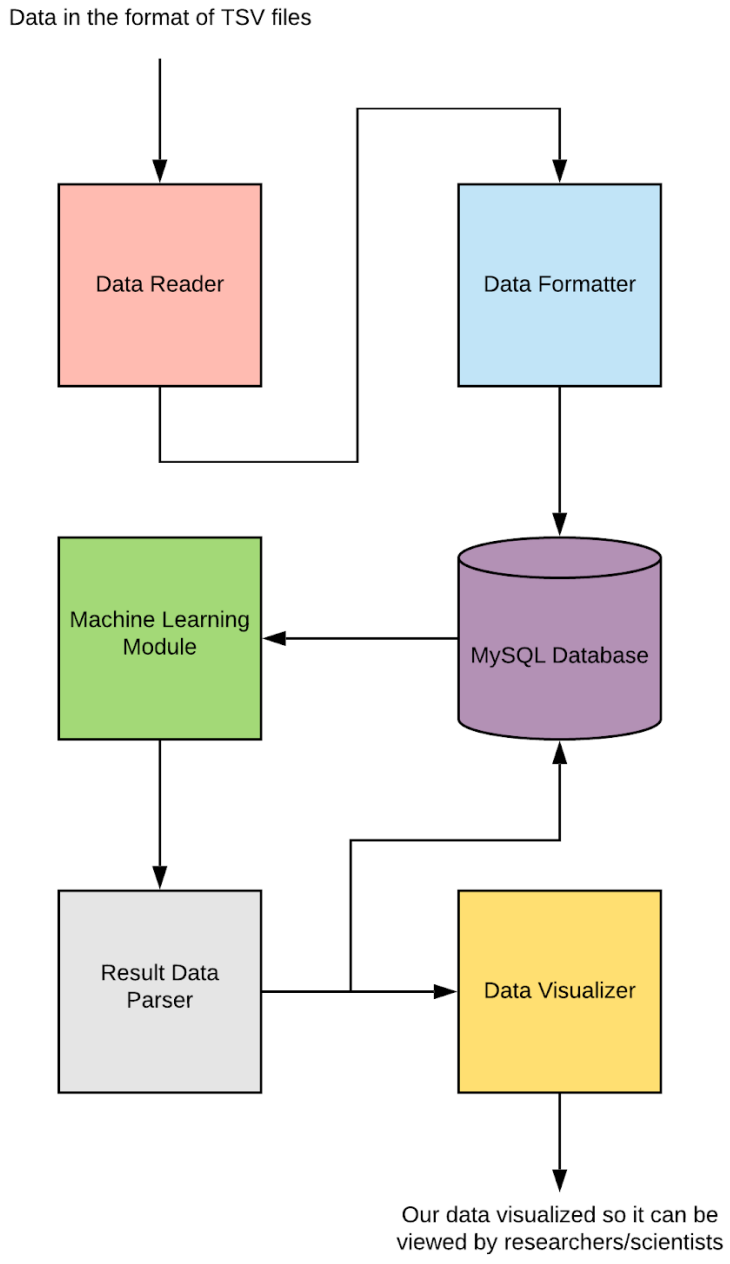
So far, we have done preliminary research into what methods we will be using to design our system. We have found that the TensorFlow library will best suit what we need. To continue with the project, we need to figure out a system to read in the data for cleaning. We will also need to do experiments with TensorFlow to become more familiar with the platform. We will also have to do more research into unsupervised learning because all of our research so far has been on supervised learning.

2.3 Development Process

For this project, we have decided to follow the Agile Development Process. The Agile Development Process is a very intuitive process and is well-known amongst everyone within our group making it great for our team to use. This process allows us to easily assign new tasks to members, monitor the tasks that are being worked on, and quickly fix any issues that may arise from any tasks. Along with those positives Agile allows us to quickly and efficiently make changes to our project as we may see fit.

2.4 Design Plan

Our design is created in a modular fashion where the data we get is put through a data reader, which then goes through a formatter so that we can parse through it much easier. This parsed and formatted data then gets stored in our database, in which then, our machine learning module will take and retrieve results from the input data. The result data will be parsed and stored in our database, as well as visualized in a friendly manner. The diagram below details the data flow and the components proposed.



3. Statement of Work

3.1 Previous Work And Literature

We will not be using previously created software, and we expect to use our own, with the exception of libraries and dependencies. There has been research on machine learning to study aging, but not using the data we will be using. We expect our project to bring new information to the world of Gerontology.

3.2 Technology Considerations

We plan on using TensorFlow which has many advantages and some disadvantages, especially when compared to its competitors:

- Advantages of TensorFlow include its graph visualization, its library management, its tools for debugging, and scalability.
- Disadvantages of TensorFlow include its lower computation speed and limited GPU support.

We plan on using Plotly as a way to visualize the results our project creates.

- Advantages of Plotly include its interactive chart, diverse set of graphs to use, and ease of use.
- Alternatives include Matplotlib, but we determined that Plotly fit our project better and was easier to use.

3.3 Task Decomposition

We expect to decompose each module into a series of tasks to complete, each with their own dependencies on other tasks. These tasks will be split up throughout the weeks in order to keep progressing towards the final project. Decomposing the tasks will also allow teammates to be able to focus on small goals/tasks to finish rather than feeling overwhelmed by a large daunting task. The large decomposition includes the deliverables for the project, which we will then break down into smaller tasks similarly to the following:

1. Inception
    a. Gathering requirements
    b. Discussion with our client
    c. Research machine learning concepts
2. Design
    a. Document project components
    b. Document component communication
    c. Create UML diagrams

3. Data parser and data formatter
    a. Parsing DDI files
    b. Parsing MIDUS TSV data files
    c. Formatting to transition to database
4. Database
    a. Set up the server for our database
    b. Create the database schema
    c. Link our project to the database
5. Machine Learning Module
    a. Set up machine learning training
    b. Test machine against new data
    c. Client approval of machine learning module
6. Result Data Parser & Visualizer
    a. Create parser to parse result data from machine learning module
    b. Create link from data parser to the database to store results
    c. Create visualizer to show the plots of results

3.4 Possible Risks And Risk Management

Our beginner's knowledge of machine learning and linear algebra affects some of our members. This is a risk that may impact the progress and timeline of our project. This may also contribute to accuracy issues in the results if we are unable to make up the time impacted. We're mitigating this risk by researching a lot on machine learning in our first semester and doing practice exercises related to the technical requirements.

Another issue that may slow down our plan is not being able to easily read in data to plug into our machine learning algorithm. To mitigate this we have been contacting other researchers to try and find a tool that can be used to read in DDI files and if that doesn't work we will be working on how we can interpret the massive CSV files to create a reader that does the work for us.

Our project relies on having virtual machines hosted by Iowa State University. If the University is unable to host the virtual machines anymore, or our virtual machines become corrupted in any way, we could lose data for our project. We are mitigating this by backing up the data we are working with and the code we are writing on both our personal computers and on our team Git repository.

3.5 Project Proposed Milestones and Evaluation Criteria

Each deliverable we have is considered a major milestone for our project. Our most important milestone will be finishing the machine learning module, as that is the main focus of our project. This milestone will be the most work-intensive module to create. In addition, each module will be thoroughly unit tested as it is being developed to ensure that it is bug-free.

Our first important milestone will be recreating a study already done on the MIDUS data set. If we are able to do that, then we will have a solid base for making additional observations about the data. This milestone will prove that both our data is filtered correctly and that our regression model works.

The project will be evaluated based on the accuracy and ability of the machine learning algorithm to interpret the data. Our client will work with us to determine whether it is meeting the standards we have set out to achieve. The results will also be evaluated on the visual presentation we display with the data. In other words, is it easy for someone not involved in the project to read and understand the results.

3.6 Project Tracking Procedures

We will use both GitLab and Jira to keep track of our progress. We will use GitLab for the low-level details, code reviews, and code changes, and use Jira for the high-level details and card completion. We will have weekly meetings where we discuss the progress that each member has made and what parts we are struggling with.

3.7 Expected Results and Validation

The desired outcome of this project is to analyze data on aging to see if there are any patterns that show up that may indicate how one can lead a healthier lifestyle. With the analyzed data we would be able to then send it off to other researchers who know more about aging and can use our data to help others make healthier decisions.

We expect we will have a fully-functioning program that can input data and output results that help understand the effects of aging. We will do extensive testing and communication with faculty members knowledgeable on the matter to confirm what we create is accurate. We also hope to introduce new data at the end to see how successful the training data was in training our module.

4. Project Timeline, Estimated Resources, and Challenges

4.1 Project Timeline

Below describes the milestones and tasks in the Gantt chart above:

Deadline - November 15, 2019: Inception

- Requirements are gathered for the project.
- Research is done on multiple machine learning concepts.
- Use of libraries and technologies are determined.

Deadline - December 20, 2019: Design

- The project is fully conceptualized and tested.
- Design of modules are created with appropriate data paths.

Deadline - January 25, 2020: Data Parser & Formatter

- We develop the data parser and formatter.
- This module is created so that linking it to the database is done with ease.
- These modules are thoroughly tested.

Deadline - February 20, 2020: Database Storage

- We set up our database so that we can store formatted input and output data.
- The data parser and formatter modules are linked to the database.
- This module is thoroughly tested.

Deadline - March 30, 2020: Machine Learning Module

- We create our machine learning module that reads in the formatted data from our database.
- The database is linked up to this module so that this module can read from it.
- This module is thoroughly tested.

Deadline - April 30, 2020: Data Visualizer & Completion

- We create the data visualizer that allows our results to be easily viewed by scientists and researchers.
- The database is linked to this module so that the results can be stored
- This module is thoroughly tested.

4.2 Feasibility Assessment

We predict that it is highly likely that our project will be finished by the end of the second semester. The biggest challenge we must overcome include our beginner's knowledge on machine learning. However, the research and practice we are doing prior to the actual production of the module will help us overcome this challenge. Another challenge I may see us running into is being able to understand the data that we are using to train our machine learning algorithm. While the data may be well documented, there is a lot of it that we have to go through so being able to piece together the different variables will be essential to figuring out how the data was collected and what it means as a whole.

4.3 Personnel Effort Requirements

Task	Hours per week	Explanation
Inception	3-5, 6+	The inception process includes gathering requirements, discussing with the client, and researching machine learning topics. These are all items that require time and focus in order to create a strong base for the understanding of machine learning and how we can use it in our project. Therefore, the items here would be described as medium- to high-complexity items.
Design	1-2, 3-5	The design task includes documenting project components, documenting component communication, and creating UML diagrams. These items are more related to discussion and overview of our project and design process. Since these items don't require a large amount of technical research or time, they are considered low- to medium-complexity items.
Data Parser and Formatter	3-5	The data parser and formatter involves parsing DDI files, parsing MIDUS TSV data files, and formatting to transition to database. These tasks are items that we have researched in the inception stage, so they should not be too time consuming. However, due to their technical nature, they are still classified as medium-complexity items.
Database Storage	3-5, 6+	The database storage includes setting up the server for our database, creating the database schema, and linking our project to the database. Due to the challenges and

		time that may come with designing a database that communicates efficiently, this task is considered medium- to high-complexity.
Machine Learning Module	3-5, 6+	Creating the machine learning module consists of setting up machine learning training, testing machine against new data, and client approval of machine learning module. Depending on how much our preparation aids us in this phase, the task of creating the machine learning module could be considered medium- to high-complexity. The item that we suspect with take the most time will be training the machine and editing the module based on our results.
Data Visualizer and Completion	3-5	The data visualizer and completion phase includes creating a parser to parse result data from machine learning module, creating a link from data parser to the database to store results, and creating a visualizer to show the plots of results. These tasks require a lot of data parsing and interpretation. We do not expect to run into many challenges while completing this task, so it is classified as a medium-complexity task.

The duration of the task depends on the complexity of it. For a low-complexity task, we expect 1-2 hours to be spent. For a medium-complexity task, we expect 3-5 hours spent. For a high-complexity task, we expect 6+ hours and plenty of technical communication with other team members to finish the task.

4.4 Other Resource Requirements

The resources required will be our own personal machines, the data that we will use for our machine learning program and the GPU clusters provided by Iowa State University. The MIDUS data set will be required for our program. Along with those we have also put to use a virtual machine to store data that we will be reading in from the different research files.

4.5 Financial Requirements

The financial requirements for this project is small if any due to the free cost of the GPU clusters and database server provided by Iowa State University, the free access to the MIDUS data sets, and the use of our own personal machines.

5. Testing and Implementation

5.1 Interface Specifications

We expect to use Mockito and the unittest framework in Python for testing.

5.2 Hardware and software

The same hardware that we use to develop will also be used for testing. All of our changes will be tested in our Iowa State virtual machine.

5.3 Functional Testing

We expect to unit test each of our modules. We will then combine our modules and test the reliability and efficiency of the system as a whole. We will present it to our client to make sure it meets standards.

5.4 Non-Functional Testing

We will do this using the GPU clusters at Iowa State University to test performance. There will be assessments on algorithms used to ensure the lowest time-complexity. We will be using the program as we develop to ensure usability.

5.5 Process

We have not been able to test our process because we have not reached that part of the project yet.

5.6 Results

We have not obtained any results because we have not reached that part of the project yet.

6. Closing Material

6.1 Conclusion

We do not have any conclusion because we have not reached that part of the project yet.

6.2 References

There are no references yet at this point of the project.

6.3 Appendices

There is no additional information yet at this point of the project.