

# Final Report

**Name:** Alex Hoffer, Chongxian Chen, Jacob Smith

**Group Number:** 59

**Project Title:** Machine Learn Your Way to March Madness Glory!

## Abstract

The application of machine learning to Biochemistry and Biophysics has enabled researchers in this field to make remarkable discoveries, such as the generation of new DNA sequences. However, students of Biochemistry and Biophysics do not get the opportunity to learn machine learning. Dr. Victor Hsu of the Oregon State University Biochemistry and Biophysics department has commissioned the Stat Champs to produce an instructional module to give his students the chance to familiarize themselves with machine learning. The software product the Stats Champs have agreed to develop is a web page that allows students to train a machine learning model based on the college basketball statistics and machine learning algorithm of their choosing in order to produce a March Madness bracket. This will help students understand how machine learning algorithms produce models and how inclusion or exclusion of certain data can influence such models. Over the course of Fall term 2016, the Stat Champs developed materials such as design documents and technology reviews in order to prepare for the engineering of the module. Then, in Winter term 2017, the Stat Champs began the software development phase of this project. In Spring term 2017, the project was finished. This report comprehensively describes the lifecycle of producing the project.

## CONTENTS

<b>1</b>	<b>Introduction</b>	3
<b>2</b>	<b>Original Requirements Document</b>	3
2.1	Changes from Original Requirements Document . . . . .	3
<b>3</b>	<b>Original Design Document</b>	3
3.1	Changes from Original Design Document . . . . .	3
<b>4</b>	<b>Original Technology Review</b>	3
4.1	Changes from Original Technology Review . . . . .	3
<b>5</b>	<b>Weekly Blog Posts</b>	3
5.1	Alex . . . . .	3
5.1.1	Fall Term . . . . .	3
5.1.2	Winter Term . . . . .	4
5.1.3	Spring Term . . . . .	5
5.2	Chongxian . . . . .	5
5.3	Jake . . . . .	5
<b>6</b>	<b>Final Poster</b>	6
<b>7</b>	<b>Project Documentation</b>	7
7.1	How the project works . . . . .	7
7.2	How to use the software . . . . .	7
7.3	Requirements for usage . . . . .	7
7.4	API Documentation . . . . .	8
<b>8</b>	<b>How We Learned New Technology</b>	8
8.1	Alex . . . . .	8
<b>9</b>	<b>What We Learned</b>	8
9.1	Alex . . . . .	8
9.2	Chongxian . . . . .	9
9.3	Jacob . . . . .	9
	<b>Appendix A: Essential Code Listings</b>	9

## 1 INTRODUCTION

Biochemistry and Biophysics are two fields that are ripe with many exciting breakthroughs. Machine learning is used by these disciplines to aid their research by providing the ability to generate new sequences of DNA. Our client, Dr. Victor Hsu, is a professor in the Biochemistry and Biophysics department at OSU who recognized that his students should understand machine learning in order to prepare them for their careers. He noticed that the department does not encourage learning machine learning, and that even if they were to, machine learning is a difficult topic for people who haven't been trained in computer science. Additionally, learning machine learning through its application to these fields can be confusing, since machine learned models of DNA can be tough to interpret. Therefore, he commissioned us to produce an online instructional module where his students could grasp machine learning fundamentals in a fun and straightforward manner. To achieve this, he asked us to provide an interface where users could select from men's college basketball statistics and generate a March Madness bracket. In doing so, budding scientists would be able to familiarize themselves with basic machine learning algorithms and also witness how the inclusion or exclusion of data influences the models trained on them. Alex Hoffer developed the GUI, Chongxian Chen implemented machine learning algorithms, and Jacob Smith gathered data for use. Dr. Hsu's role in the project was supervision only. This report chronicles the lifecycle of the project through Fall, Winter, and Spring terms in 2017.

## 2 ORIGINAL REQUIREMENTS DOCUMENT

### 2.1 Changes from Original Requirements Document

## 3 ORIGINAL DESIGN DOCUMENT

### 3.1 Changes from Original Design Document

## 4 ORIGINAL TECHNOLOGY REVIEW

### 4.1 Changes from Original Technology Review

## 5 WEEKLY BLOG POSTS

### 5.1 Alex

#### 5.1.1 Fall Term

5.1.1.1 Week 3: This week we collaborated with Dr. Hsu and wrote our Project Statement, which he reviewed and signed off on. We submitted the Project Statement at 11 am on 10/14/2016. We did not really encounter any problems, except for maybe some LaTek formatting issues. I am sure the use of LaTek will become easier with more practice. Next week we will be submitting our resumes for peer review and figuring out how to proceed with this project now that we have a clear, working vision to follow.

5.1.1.2 Week 4: This week we finalized our project statement by revising it to reflect our instructors' suggestions. We also all produced resumes and gave them to classmates for feedback. Next week we will work on our project requirements document and attend Career Fair.

5.1.1.3 Week 5: This week we attended career fair and developed our project requirements document. Next week we will continue developing this document and will consult the client to get his approval.

5.1.1.4 *Week 6:* This week we revised the rough draft of our project requirements document, formatted it using LaTek, and added a Gantt chart. We submitted this requirements document to our client, but as of 3:20 pm have not heard back from him. By the end of the weekend, we hope to have the document signed and submitted. Next week, we will individually work on our tech reviews.

5.1.1.5 *Week 7:* This week we revised our project requirements document and began working on our tech review document. Next week we will finish our tech review document.

5.1.1.6 *Week 8:* This week we developed our technology review document. We ran into some issues coming up with 3 responsibilities for everybody. We also had some difficulty identifying potential technologies for each of these responsibilities. Next week we will complete our design document.

5.1.1.7 *Week 9:* This week we talked about how we would make our design document, and discussed how we would approach recording ourselves for the progress report. We are having some difficulties with design document formatting. We will probably rent out a microphone for use with a computer to record ourselves for the progress report. Next week we will turn in our design document.

5.1.1.8 *Week 10:* This week we finished our design document. We will send it to our client and get a signature as soon as possible. We faced difficulty in getting LaTex to properly generate designs like message sequence diagrams. Next week we will submit our progress report and conclude the term.

## 5.1.2 Winter Term

5.1.2.1 *Week 1:* This week we came back from winter break and re-calibrated. We voted on a meeting time (Tuesdays) and attended the first class. We look forward to the term.

5.1.2.2 *Week 2:* This week we had our first meeting back with our TA. Our meeting consisted of planning out the term for our team. This meant we re-established the responsibilities we set for ourselves individually last term and verbally sketched out an idea of what our Beta release would look like. My own contributions this week were setting up the web page the module will be hosted on and doing some GUI work. The web page can be found here: <http://web.engr.oregonstate.edu/hoffera/CapstoneProject/MachineLearnYourWayToMarchMadnessGlory.html>. More to come on the GUI work. Chongxian has set up some machine learning algorithms we can use so next week we will have Jake provide the data to them to see how they operate. After we get a handle on these algorithms, we will begin setting up the module.

5.1.2.3 *Week 3:* This week Chongxian arranged the machine learning algorithms, Jake compiled the statistics we'd use in a .csv file, and I worked on the GUI of our web page. Next week we have class on Thursday and we should have the algorithms being allowed to accept stats as input.

5.1.2.4 *Week 4:* This week I continued to polish the GUI for the webpage. Chongxian has selected our machine learning algorithms and Jake has helped him find examples of how to implement them. Jake also gathered a lot of basketball statistics for use in the module. We need to do the OneNote portfolio, a progress report, and a voice-over update by late February.

5.1.2.5 *Week 5:* This week we attended class, Chongxian continued developing our machine learning algorithms, Jake continued to gather data, and I continued to develop our GUI. Next week we plan to release an alpha version of our module and we need to create a OneNote, edit our documents, make a status report, and submit these to the OneNote.

5.1.2.6 *Week 6:* This week we completed our progress report, both written and presentation versions. I made our OneNote and uploaded all of our documents to it. We had a bit of a hard time filling up all of the required time for

the presentation. Next week we will continue development.

5.1.2.7 *Week 7:* This week we continued coding. I re-submitted my OneNote to Dr. Winters because it didn't go through the first time. I also met with Dr. Winters to modify my OneNote a bit. We need to finish our coding to be at a beta level release.

5.1.2.8 *Week 8:* This week I waited for Jake and Chongxian to make progress. Next week we need to setup a meeting with Dr. Hsu. We also are supposed to be presenting a beta release of our project.

5.1.2.9 *Week 9:* This week we did elevator pitches. I have made a draft of our poster using a LaTex conference poster template, and I added a signature page to our progress report and sent it to our client. My two partners should have made progress on integration. Next week I need to polish the poster more and write my new progress report, which I will use the IEEEtran format for.

### **5.1.3 Spring Term**

5.1.3.1 *Week 1:* This week I went to class to get accustomed to what the term would entail. I updated some CSS on our page. Then, I emailed our client with a link to our project. Hope to hear from him next week and hope to update our poster for Expo.

5.1.3.2 *Week 2:* In week 2, Chongxian implemented different algorithms to choose from. We all updated the poster and submitted our new draft. We also emailed our client again with our finished product and our draft for him to sign off on. We await his response. In the coming weeks, we need his sign-off on the poster and then we have Expo.

5.1.3.3 *Week 3:* This week we attended class. I went to Dr. Hsu's office to talk to him about signing off on our posters. We also pushed our code to the Github repo. Our poster final is due May 1.

5.1.3.4 *Week 4:* This week I modified our poster to match our client's suggestions, then McGrath's suggestions. Then, I submitted the poster for printing. Next week I need to do the WIRED assignment.

5.1.3.5 *Week 5:* This week I interviewed Brandon Chatham for the WIRED assignment. We are preparing for Expo.

5.1.3.6 *Week 6:* This week we got our spring term progress report ready. Expo is next week and we're preparing.

5.1.3.7 *Week 7:* This week we had Expo. Term's almost over. In the remaining three weeks we need to edit our original docs, finish 3 small writing assignments, and do a final progress report.

## **5.2 Chongxian**

## **5.3 Jake**

## 6 FINAL POSTER

**COLLEGE OF ENGINEERING**

**Introduction/Background**

**IMPORTANCE OF MACHINE LEARNING TO BIOCHEMISTRY AND BIOPHYSICS**

Biochemistry and biophysics are two fields that are ripe with many exciting breakthroughs. Machine learning, a type of artificial intelligence where computer programs adapt to new data, is used by biochemists and biophysicists to do things like analyze genomic DNA sequences. Our client, a professor in the department of Biochemistry and Biophysics at OSU, recognized there was a need for his budding scientists to understand machine learning so they could be better prepared for their careers.

**NEED FOR A MACHINE LEARNING INSTRUCTIONAL TOOL**

Our client noticed that the Biochemistry and Biophysics curriculum at OSU did not encourage undergraduate students to learn machine learning. Even if machine learning classes were to become a cornerstone of their coursework, the content would be difficult for people without a Computer Science background. To make matters worse, teaching machine learning to these students through its application to biochemistry is particularly challenging, since biochemical results are non-definitive in that DNA sequences often do not need to be exact and can be unclear. Meanwhile, college basketball results are win-lose and therefore it is more straightforward to interpret the differences in results based on changing the inputs.

**WHAT WE WERE COMMISSIONED TO DO**

We were enlisted to produce an online instructional module where these students could grasp machine learning fundamentals in a clear manner. Our client wanted us to develop this module so that students could generate machine learned NCAA March Madness brackets. Since a fundamental aspect of learning machine learning is recognizing how the inclusion or exclusion of data influences resulting models, this module would satisfy the need by producing models (brackets) that were distinguishable from each other based on the college basketball statistics a user chose for training.

**Fig. 1: Home page of website that includes project information and a link to our module.**

**Electrical Engineering & Computer Science**

# MACHINE LEARN YOUR WAY TO MARCH MADNESS GLORY!

## Teaching Biochemists and Biophysicists Machine Learning

**2. AGGREGATE/SELECT STATISTICS**

Jacob gathered college basketball statistics from 1985 to the current season from the website Kaggle.com in the CSV file format. Since the regular season didn't conclude until March, Jacob manually updated the database to reflect the current standings frequently until the final game was played. Then, he added stats from the tournament for future use in algorithms and analysis. We used a Python script to allow users to choose from a wide variety of stats including categories like field goals attempted per game to train a model on, as demonstrated by Fig. 2.

**3. FEED STATISTICS TO MACHINE LEARNER**

Using the Python Scikit-Learn library, Chongxian read the CSV files of the user selected statistics into Numpy arrays.

**4. TRAIN A MODEL USING AN ALGORITHM**

Along with their choice of statistics, users are also able to choose between different machine learning estimators such as Linear Regression and SVM Polynomial. By using a basketball ELO rating system, the supervised machine learning model is able to fit on the statistics and predict new matches. A CSV file of the match results between two teams with the probability is generated as a result. The bracket results effectively present how the user's choice affects the machine learning prediction.

**5. GENERATE BRACKET OF RESULTS**

While a machine learned module is being generated, the user is presented with a screen that includes the command line arguments given to Scikit and informs the user on which steps are necessary to complete their request. The prediction CSV file generated from the machine learning model was then transferred into bracket form by Jake using a Python script.

**Fig. 2: Menu where the user chooses a machine learning algorithm and stats to generate a bracket with.**

**Fig. 3: A March Madness bracket predicted by the SVM RBF algorithm.**

**CONCLUSION**

Alex Hoffer, Jacob Smith, Chongxian Chen

**FEATURES PROVIDED BY THE MODULE:**

- A Graphical User Interface (GUI) including a "Home" page, "Instructions" page, "About The Developers" page, and a "Purpose" page.
- The ability to select from a set of college basketball statistics.
- The ability to select from a set of popular machine learning algorithms.
- A machine learned bracket that corresponds to the specific statistics and algorithm the user requested the model to be trained on.

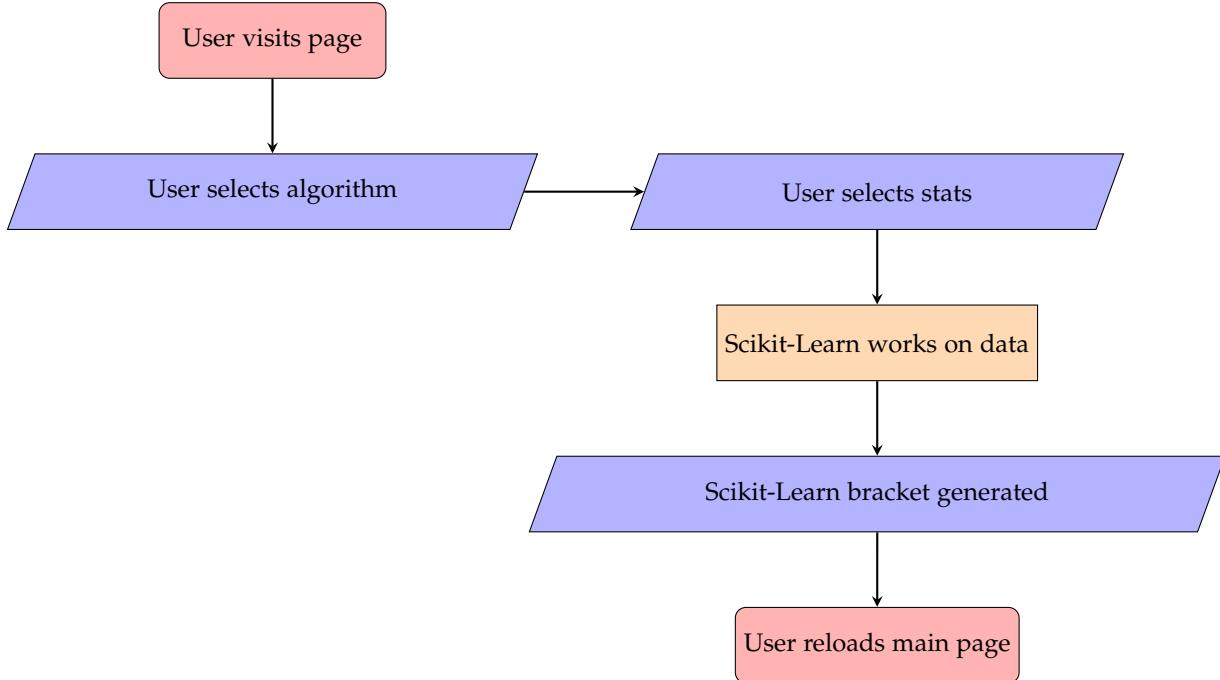
The project was completed in early April. All functional requirements as outlined by our client were completed. Future improvements to our module could include more machine learning algorithms, a wider variety of statistical categories, a more elegant looking outputted bracket, and finding a way to increase the speed at which the machine learning algorithms generate results. Additionally, the developers of such modules may wish to have prior machine learning experience, more proper modes of communication, and a more specific work schedule established before development in order to allow for time after completion to polish each component of the module separately.

**Oregon State UNIVERSITY**

## 7 PROJECT DOCUMENTATION

We were both lucky and unlucky to be assigned a project that did not make well-detailed documentation a necessary responsibility. We were lucky in the sense that this shaved off a lot of work we would have to do, and unlucky in the sense that this project did not expose us to how to properly write documentation.

### 7.1 How the project works



### 7.2 How to use the software

No installation is necessary. To use our software, all a user has to do is go to this website: <http://web.engr.oregonstate.edu/hof-fera/CapstoneProject/MachineLearnYourWayToMarchMadnessGlory.html>. On this site, a link to the machine learning module is posted. The module is compatible across all browsers. There are instructions on how to use the module on this site, as well as on the page where the module is hosted itself. To use the module, the user clicks on one machine learning algorithm and any number of the provided statistical categories and submits their choices. Then, a waiting period ranging from two minutes for simple machine learning algorithms to many hours for more precise algorithms is required. A message displays on the screen to inform the user they must wait. Then, the user is informed when the bracket has been generated. Finally, the user returns to the module page to see their generated bracket presented. The user can repeat this process an arbitrary number of times to generate an arbitrary number of brackets, but of course, the amount of time it takes to utilize certain machine learning algorithms is an important constraint the user must consider.

### 7.3 Requirements for usage

No special hardware, operating system, or runtime requirements dictate the usage of this module. As stated previously, any browser on any operating system with internet access can utilize the module.

## 7.4 API Documentation

We were not asked to document any API, and do not have any user guides. Such documents would not be particularly useful to any programmers because our module uses Scikit-Learn. This means that the API has already been documented for us by the fine people at Scikit-Learn. This documentation can be found here: <http://scikit-learn.org/stable/documentation.html>.

# 8 HOW WE LEARNED NEW TECHNOLOGY

## 8.1 Alex

### 8.1.0.8 Websites:

- <https://stackoverflow.com/>
- <https://www.w3schools.com/js/>
- <http://scikit-learn.org/stable/documentation.html>

Stack Overflow was most valuable because it provided useful information on debugging small issues with web development. W3 Schools was second to this, because it gave me a lot of insight on how to use JavaScript to make a website *pop*. JavaScript is one of my weaker languages, so I needed it to make sure I was doing the right things. Finally, the documentation for Scikit-Learn was of course valuable for the inclusion of machine learning algorithms, but is listed last because implementation of machine learning algorithms was primarily Chongxian's responsibility.

8.1.0.9 *People*: Several people on campus were tremendously helpful in providing information that we needed. We owe a debt to our teaching assistant Xinze Guan for recommending Scikit-Learn for machine learning algorithm implementation. Our client, Dr. Victor Hsu, was also useful in this regard. We also relied on our instructor Kevin McGrath's LaTeX templates and Makefiles and Dr. Kirsten Winters' writing advice. Finally, my teammates helped me learn a lot of new and exciting technologies, with Jake demonstrating to me proper usage of LaTeX and Chongxian for exploring Scikit-Learn and elucidating a lot of its mysteries to me.

# 9 WHAT WE LEARNED

## 9.1 Alex

9.1.0.10 *Technical Information*: In terms of web development, I learned more JavaScript and polished my HTML/CSS abilities. I became acquainted with how Amazon Web Services works and what their limits on their free service are. I also learned a little on how to write embedded Python, how to use Scikit-Learn in Python, and how to use Python to scrape data from websites. Since these were not my primary responsibilities, though, I didn't learn them as well as I would've liked. Finally, I learned how to use LaTeX, proper usage of the IEEETran format, and the mathematical underpinnings of various machine learning algorithms.

9.1.0.11 *Non-Technical Information*: I learned how to interact with a software client. I also learned how to take an idea and move it through the necessary phases of development, including requirements, planning, and design phases all the way to implementation, maintenance, and presentation. Essentially, I learned the engineering practices necessary to develop a project from embryo to adulthood, and learned how to sell or pitch an idea.

9.1.0.12 *Project Work*: I learned that a project must run on a well-defined schedule. If each milestone of a project builds on a previous one, this is especially important. Without a rigid schedule, you fall behind sooner rather than later and the versions of your project you will release will be fraught with bugs.

9.1.0.13 *Project Management:* I learned that in a team setting, somebody has to be the team manager (whether de facto or not), otherwise milestones won't be reached and the project will be delivered late.

9.1.0.14 *Working in Teams:* I discovered certain methods for overcoming differences in work style. Some people procrastinate, others get work done early. There's nothing wrong with either approach, but if you're in a team where there are conflicting philosophies, you need to find out early how to bridge the gap between you and your group mates.

9.1.0.15 *What I would do differently:* I would choose different responsibilities for this project. I wish I could've done more of the machine learning aspects, rather than simply oversee Chongxian's progress. I would've considered using something besides Scikit-Learn because I would've liked to have used more exotic machine learning algorithms, or at the very least neural networks, which are not supported in Scikit-Learn.

## 9.2 Chongxian

## 9.3 Jacob

## APPENDIX A

## ESSENTIAL CODE LISTINGS