Spring Term Progress 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 eld 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 inuence 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 process of finishing up the project.

CONTENTS

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. This report chronicles the progress of our team through our final term.

2 ALEX

2.1 Where Alex is At On the Project

Alex completed his requirements early in the winter term. Thus, the only things he can do now are to make the module look nicer, though any changes made now won't be included in the product version that we will be presenting at Expo on May 19, because it would run the risk of introducing bugs. Since our current module satisfies all functional requirements, it would be unwise to add code that could meddle with its performance. It may be nice to format the bracket that is generated in a way that is more appealing, colorful, and relevant. Perhaps this could mean adding a cool March Madness logo somewhere. If we were to polish the module more, while it would not add any *pizzazz* to the Expo version, it would make our project more marketable to potential employers.

2.2 What Alex Has Left To Do

The remaining parts of this class that Alex has left to do are: present at Expo, turn in the spring term progress report, and complete the writing assignments needed at the end. There are no remaining things for Alex to do on the project in terms of *required functionality*. Everything that was asked by our client has been completed. If he wants to, he can polish the bracket output to make it look more presentable. Currently, it does look a bit boring, which is unfortunate for a module that does something so interesting.

2.3 Problems that have impeded my progress

I am reasonably well acquainted with web development and so my responsibilities did not pose me much grief. However, my JavaScript ability is somewhat lacking. So, I would say that the real problems I ran into in fulfilling my tasks were my neophytic JavaScript skills. One concrete issue that I ran into that I still need to resolve is how the web page appears on different browsers. The computer I used to do my development runs on Windows and I use a Chrome browser. But, when my partner pulled up the web page on a computer running an Apple operating system in a Safari browser, the divisions between the subpages of the site became unset and looked disorganized. This could be a problem because many people use Apple computers and I don't want my hard work in a high quality user experience

to go to waste just because of a lack of empathy for running software on different platforms. I also noticed that even on my browser there remains a few spots where the organization is slightly off, for example, on my group description page the text occupies too much space and is to close to the button on the bottom. I want to make this text smaller so that it does not crowd the user's vision and distract from proper navigation through the module. Additionally, I used to have a banner across the top that allowed users to select which part of the project they wanted to visit, but I changed it to a small menu button that provides this service instead. I aim to change it back. Sadly, I will have to make these changes to the product that won't be shown at Expo. In terms of the whole project, we did run into some issues with the machine learning module, particularly the fact that the embedded Python necessary to generate brackets didn't work on the OSU server. Chongxian solved this problem by switching us over to Amazon Web Services (AWS). One more intangible issue we ran into over the course of the whole project was that it was unclear when different parts of the project should be done. A lot of parts of the project depended on other parts, so it was impossible to complete them without completing the others. Since the project is finished, there's no way to change that now, but in future projects, it would be nice to run on a more rigidly defined schedule.

2.4 Particularly interesting pieces of code

```
1  <div class="col-lg-4 text-center">
2    <i class="fa fa-envelope-o fa-3x sr-contact"></i>
3    <p><a href="mailto:hoffera@oregonstate.edu">Contact Alex Hoffer</a></p>
4  </div>
```

Listing 1. HTML code that uses an icon class to load an envelope icon that links to my email

2.5 Images from my parts of the project

A more in-depth description of the images I'll be including:

1) A home page, found in *Fig. 1*. It includes a a bold title of our project, a photo of an OSU basketball player driving on a UCLA player, a button that changes color when you hover over it, and a menu that upon a click drops down to reveal links to the remaining sections of the page.

2) An instructions page, found in *Fig. 2*, that includes instructions on how to use the module.

3) A module page, located in *Fig. 3*, where the machine learning module hosted on AWS is linked to. Naturally, we were going to host it on this page itself (on OSU's server), but the server didn't support what we wanted to do.

4) A purpose page, represented by *Fig. 4*, where I wrote a short description of why the project was made. Underneath this text lies a button which takes the user to the next section, where I describe our development team and our project information. As described in the "Past Problems" subsection, I need to scrunch up this text to make it all appear without making the page a little disorganized.

5) An About the Authors page, found in *Fig. 5*. Here I include information about us and a cool email shaped button with the text "Contact Alex Hoffer" underneath it. If the email shaped button is pressed, it executes the script "mailto:hoffera@oregonstate.edu", which is my email. In my "Particularly interesting pieces of code" section, I have included the recipe for invoking this cool button in HTML.
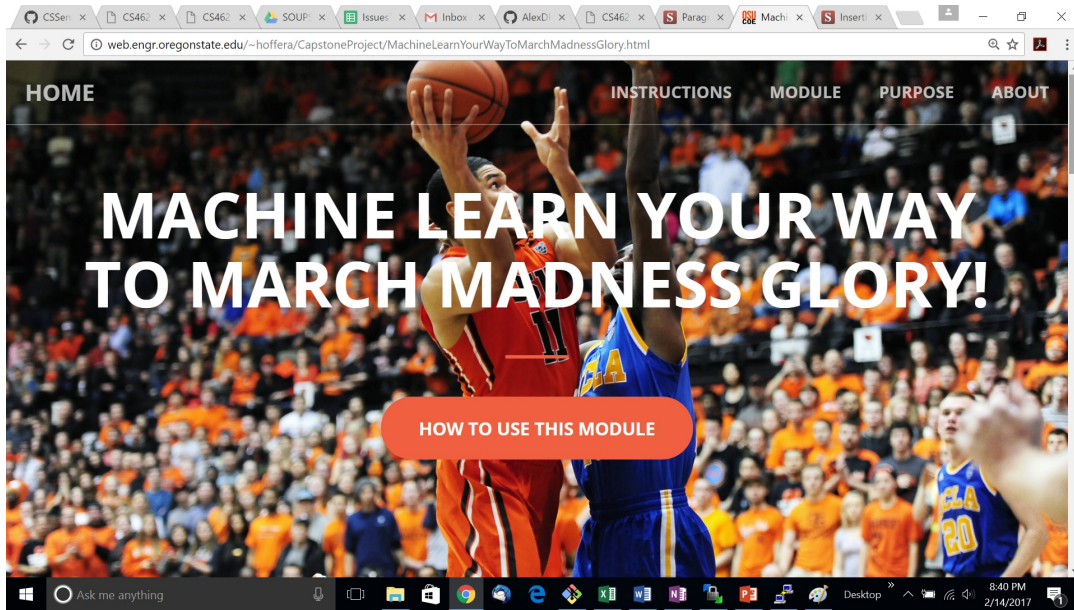
Fig. 1. A home page where the user can go anywhere in the project using "Menu" or go immediately to the next section by clicking the button.
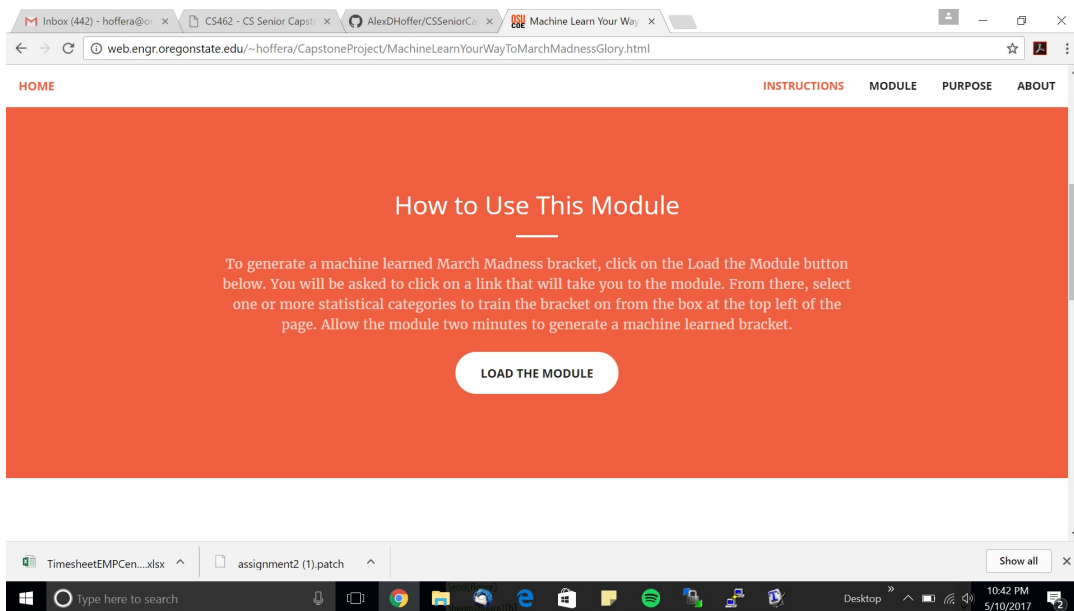

Fig. 2. Instructions on using the module presented before the module itself is presented, as outlined by Responsibility #2.
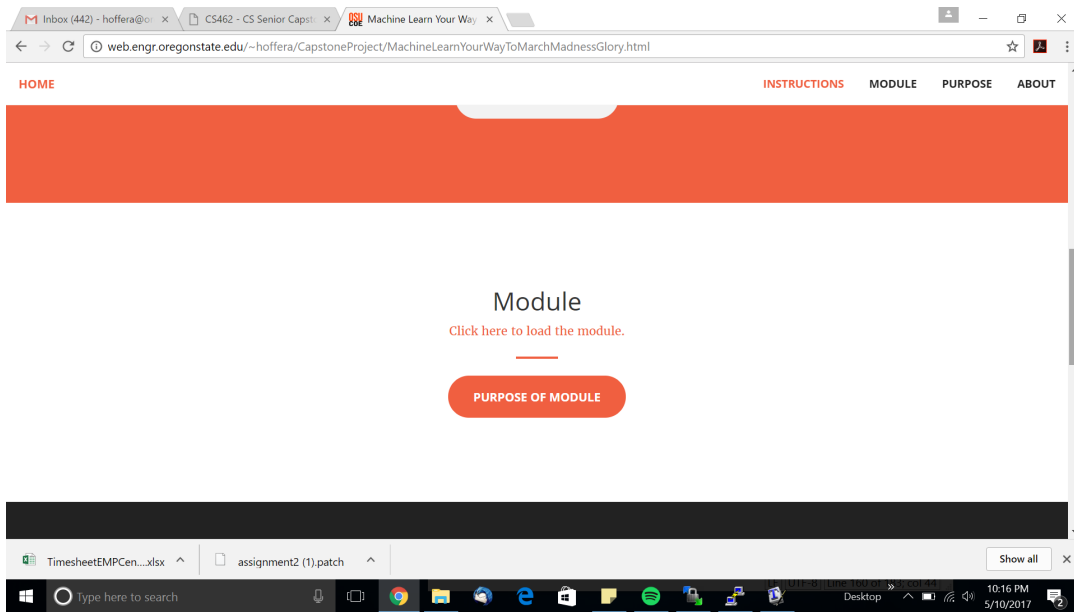
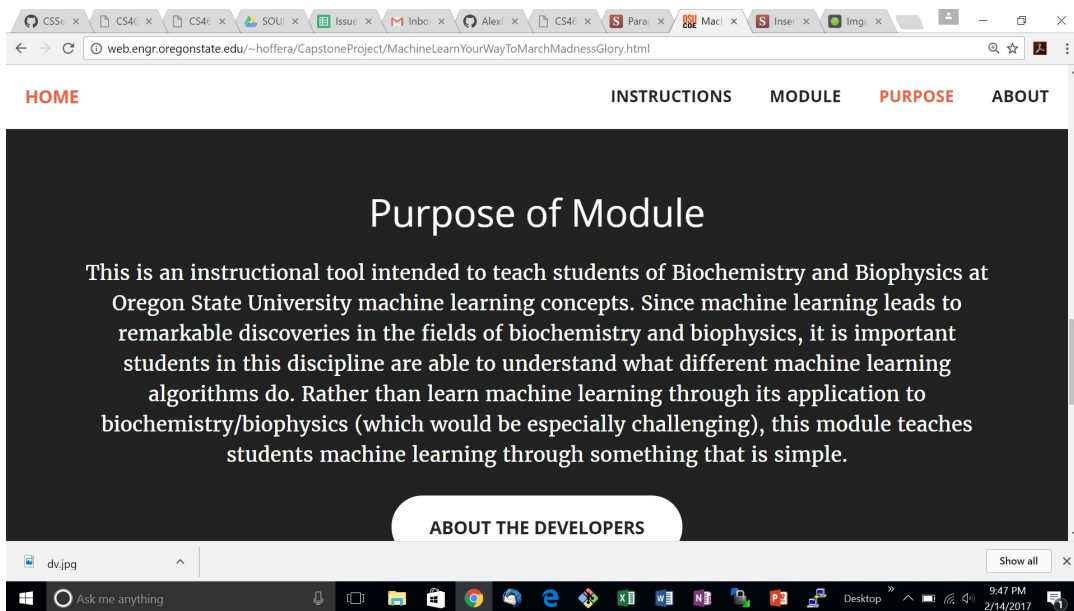Fig. 3. A link to the module hosted on AWS.

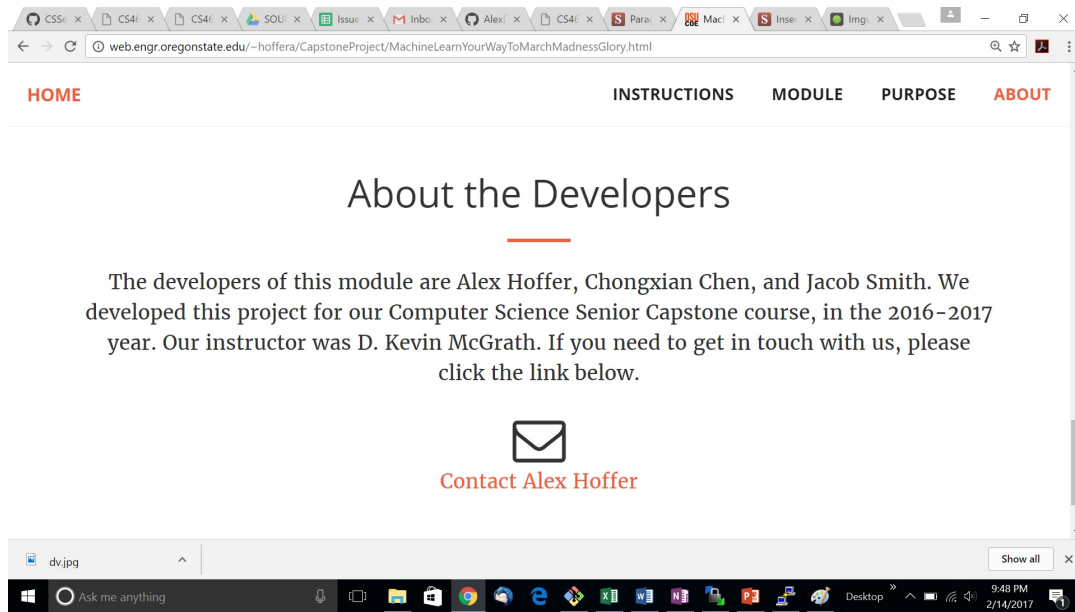Fig. 4. A page that describes why we did this project.

Fig. 5. This page describes who we are and includes a way to contact me.

3   CHONGXIAN

3.1   Where Chongxian is At On the Project

Chongxian has completed all functional requirements of the Machine Learning module including different algorithmn and different basketball stats for the users to choose from. The module is also put on Chongxian's Amazon web space so the users are able to run the python module remotely and generate the visible results.

3.2   What Chongxian Has Left To Do

The remaining parts of this class that Chongxian has left to do are: present at Expo, turn in the spring term progress report, and complete the writing assignments needed at the end. Since all the functional requirements are completed, all I can do is to improve the performance even more. The most significant thing I can do is upgrade the Amazon Web Service so users can get their results faster remotely, but that increases the cost. Other things I can do may include introducing more algorithmns to make our module more educational.

3.3   Problems that have impeded my progress

At first since none of our team members have taken Machine Learning classes before, I have to explore about it. In the beginning we don't even know what language to use or what library to start with. Our TA Xinze Guan got us started with SciKit Learn library in Python. After spending a long time reseaching, learning and modifying our script, the functional module is completed. During the process I got a lot helps from the website Kaggle.com and Jake. Then we found that OSU server doesn't run Python script. Then I moved to Amazon Web Service (AWS). But the AWS free machine has too samll memory so I spend a while figuring out how to install all the required Python environments for our project. Visualizing our results also met dificulty on AWS since there is no monitor part on AWS server, it purely is a computing machine. Thus PIL in Python got some trouble. After troubleshooting I finally figure it out and our module is successfully ran on it.

3.4 Particularly interesting pieces of code

```
1   if len(sys.argv) > 1:
2       if sys.argv[1] == '1':
3         print("You choose linear logistic regression to be the model")
4         model = linear_model.LogisticRegression()
5       elif sys.argv[1] == '2':
6         print("You choose svm rbf to be the model")
7         model = svm.SVC(gamma=0.001, C=100., probability=True, kernel='rbf')
8       elif sys.argv[1] == '3':
9         print("You choose svm linear to be the model")
10        model = svm.SVC(gamma=0.001, C=100., probability=True, kernel='linear')
11      elif sys.argv[1] == '4':
12        print("You choose svm poly to be the model")
13        model = svm.SVC(gamma=0.001, C=100., probability=True, kernel='poly')
14      elif sys.argv[1] == '5':
15        print("You choose svm sigmoid to be the model")
16        model = svm.SVC(gamma=0.001, C=100., probability=True, kernel='sigmoid')
17        elif sys.argv[1] == '6':
18            print("You choose Niave Bayes Gaussian to be the model")
19            model = GaussianNB()
20        elif sys.argv[1] == '7':
21            print("You choose Niave Bayes multinomial models to be the model")
22            model = MultinomialNB()
```

Listing 2. Python command line arguments that allow users to use different algorithmns from SciKit Learn

3.5 Images from my parts of the project

A more in-depth description of the images I'll be including:

1) The table in *Fig. 6*. The first selection includes 7 algorithms and the second selection include 10 basketball stats where users can choose multiple to generate their prediction.

2) The terminal output when running the Python Script, found in *Fig. 7*, shows the user the detail when running the script like how many samples are used and model selection score.

3) The visualized bracket in *Fig. 8*, shows users their prediction of March Madness 2017

Fig. 6. HTML table to allow users to choose algorithmns and stats

```
python mm.py
python mm.py score fga fga3 3pp ast

You chose  ['score', 'fga', 'fga3', '3pp', 'ast']  as the feature to be considered
Building season data.
Fitting on 20848 samples.
Doing model selection.
0.714120553841
Getting teams.
Predicting matchups.
Writing 2278 results.
Outputting readable results.
Generating pictures.
```

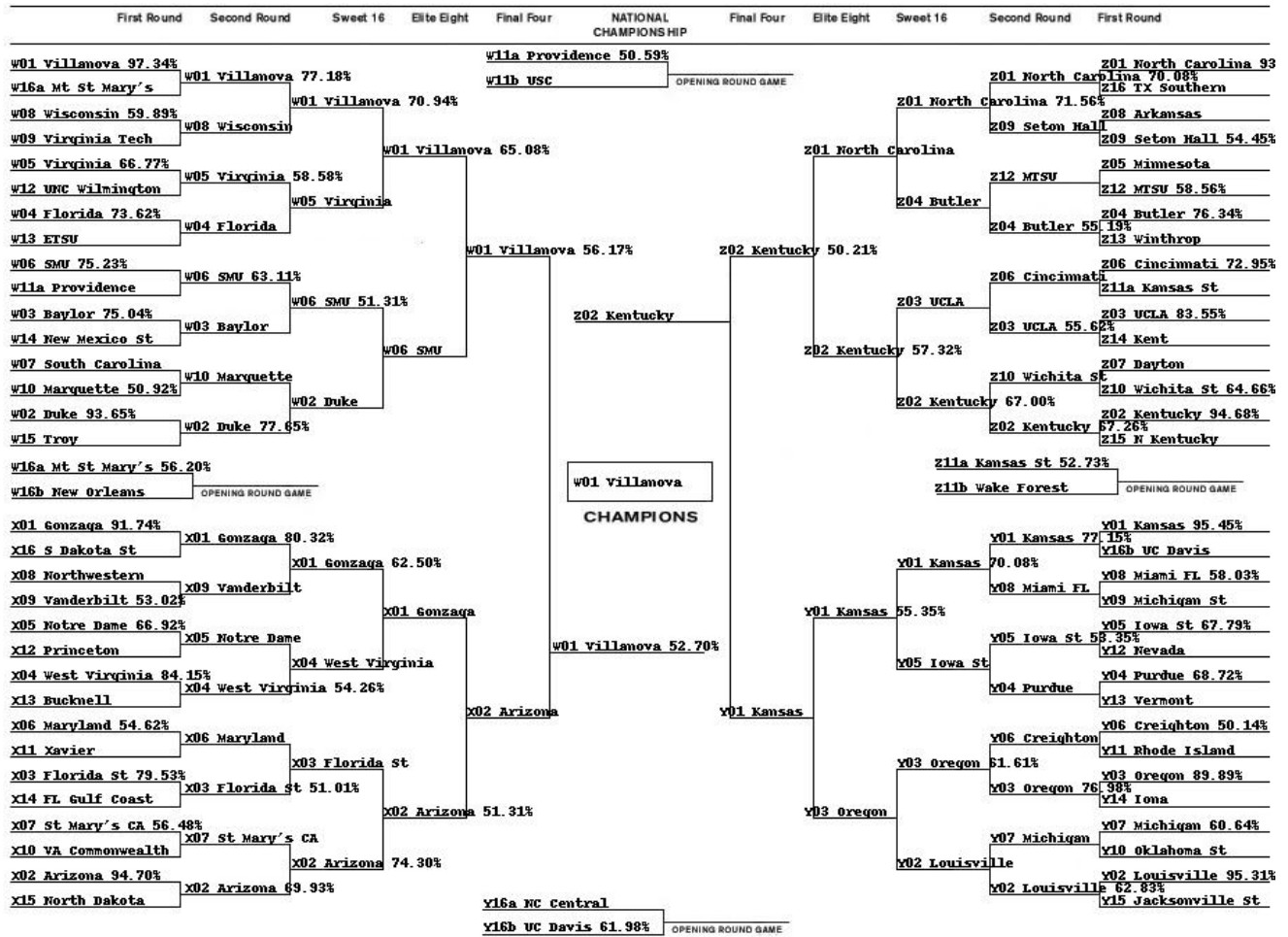Fig. 7. The terminal output when executing the Puython script

First Round   Second Round   Sweet 16   Elite Eight   Final Four   NATIONAL CHAMPIONSHIP   Final Four   Elite Eight   Sweet 16   Second Round   First Round

W11a Providence 50.59%
W11b USC   OPENING ROUND GAME

W01 Villanova 97.34%
W16a Mt St Mary's
W01 Villanova 77.18%
W08 Wisconsin 59.89%
W09 Virginia Tech
W08 Wisconsin
W01 Villanova 70.94%
W05 Virginia 66.77%
W12 UNC Wilmington
W05 Virginia 58.58%
W04 Florida 73.62%
W13 ETSU
W04 Florida
W05 Virginia
W01 Villanova 65.08%
W06 SMU 75.23%
W11a Providence
W06 SMU 63.11%
W03 Baylor 75.04%
W14 New Mexico St
W03 Baylor
W06 SMU 51.31%
W07 South Carolina
W10 Marquette 50.92%
W10 Marquette
W06 SMU
W02 Duke 93.65%
W15 Troy
W02 Duke 77.65%
W02 Duke
W01 Villanova 56.17%

W16a Mt St Mary's 56.20%
W16b New Orleans   OPENING ROUND GAME

Z01 North Carolina 93
Z16 TX Southern
Z01 North Carolina 70.08%
Z08 Arkansas
Z09 Seton Hall 54.45%
Z09 Seton Hall
Z01 North Carolina 71.56%
Z05 Minnesota
Z12 MTSU 58.56%
Z12 MTSU
Z04 Butler 76.34%
Z13 Winthrop
Z04 Butler 55.19%
Z04 Butler
Z01 North Carolina
Z06 Cincinnati 72.95%
Z11a Kansas St
Z06 Cincinnati
Z03 UCLA 83.55%
Z14 Kent
Z03 UCLA 55.62%
Z03 UCLA
Z07 Dayton
Z10 Wichita St 64.66%
Z10 Wichita St
Z02 Kentucky 94.68%
Z15 N Kentucky
Z02 Kentucky 67.26%
Z02 Kentucky 67.00%
Z02 Kentucky 57.32%
Z02 Kentucky 50.21%
Z02 Kentucky

Z11a Kansas St 52.73%
Z11b Wake Forest   OPENING ROUND GAME

W01 Villanova 52.70%

W01 Villanova
CHAMPIONS

X01 Gonzaga 91.74%
X16 S Dakota St
X01 Gonzaga 80.32%
X08 Northwestern
X09 Vanderbilt 53.02%
X09 Vanderbilt
X01 Gonzaga 62.50%
X05 Notre Dame 66.92%
X12 Princeton
X05 Notre Dame
X04 West Virginia 84.15%
X13 Bucknell
X04 West Virginia 54.26%
X04 West Virginia
X01 Gonzaga
X06 Maryland 54.62%
X11 Xavier
X06 Maryland
X03 Florida St 79.53%
X14 FL Gulf Coast
X03 Florida St 51.01%
X03 Florida St
X07 St Mary's CA 56.48%
X10 VA Commonwealth
X07 St Mary's CA
X02 Arizona 94.70%
X15 North Dakota
X02 Arizona 69.93%
X02 Arizona 74.30%
X02 Arizona 51.31%
X02 Arizona

Y01 Kansas 95.45%
Y16b UC Davis
Y01 Kansas 77.15%
Y08 Miami FL 58.03%
Y09 Michigan St
Y08 Miami FL
Y01 Kansas 70.08%
Y05 Iowa St 67.79%
Y12 Nevada
Y05 Iowa St 53.35%
Y04 Purdue 68.72%
Y13 Vermont
Y04 Purdue
Y05 Iowa St
Y01 Kansas 55.35%
Y06 Creighton 50.14%
Y11 Rhode Island
Y06 Creighton
Y03 Oregon 89.89%
Y14 Iona
Y03 Oregon 76.98%
Y03 Oregon
Y03 Oregon 51.61%
Y07 Michigan 60.64%
Y10 Oklahoma St
Y07 Michigan
Y02 Louisville 95.31%
Y15 Jacksonville St
Y02 Louisville 62.83%
Y02 Louisville
Y01 Kansas

Y16a NC Central
Y16b UC Davis 61.98%   OPENING ROUND GAME

Fig. 8. Visualization of our final result as a bracket

4   JACOB