

Modeling Physical Phenomenon via Machine Learning

ZACHARY TANENBAUM, DENIS VIRTOV*

New York University

zachtanenbaum@gmail.com, dv989@nyu.edu

May 18, 2018

Abstract

We'll do this part last! Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

I. INTRODUCTION

Outline for the introduction. Explain the goal of the project. The reasoning we chose this topic. Why it is important. Why machine learning is a good answer to our problems. (exp. Simulation is extremely expensive for some tasks, saved time and money viewing from observations instead of deriving solutions [assuming sparse learning works])

Explain that we approached the problem by using Minecraft as an "unknown" physical universe. (Drag, Vo of the Arrow, Initial y position of the arrow [1.62m higher than the block the player is standing], gravitational constant) Explain the coordinate system of minecraft (negative pitch, positive z is south [double check that], etc). Explain the units of measurement (block = 1 meter) and time (tick = 1/20th Second). Explain what the heck is projectile motion.

While we know the various hidden metrics of projectile motion in Minecraft the machine learning model does not and will have

to learn these parameters. Explain that we used project Malmo to simplify the communication between our models and Minecraft.

Our goal for this project was to come up with a machine learning model that could learn the underlying equations for projectile motion. Basic projectile motion is quite simple however once drag is introduced, the equations become quite complicated and can vary in structure depending on the conditions. We wanted to see if a machine learning model could learn these solutions. In addition, attempting to learn both the necessary force (or initial projectile velocity) and pitch in order to hit a target and clear obstacles becomes a complicated set of trigonometric equations and inequalities that are not consistent.

To accomplish this we used Minecraft as an "unknown" physical environment and trained various machine learning models on observations of player position, target position, the environment around the player, and the resulting pitch and force necessary to shoot an arrow at the target.

Minecraft's environment is a simplified ver-

*A special thanks to our adviser Vitaly Kuznetsov

sion of our own universe. Minecraft is a voxel-based game made of blocks that are 1 meter on each side. Time in Minecraft consists of ticks with 1 tick = $\frac{1}{20}$ th of a second. All physical laws are based off of these 1 meter blocks and ticks. The gravitational constant is 0.05 blocks per tick and drag slows projectiles by 1% per tick. The coordinate plane is also upside down with negative pitches aiming above the horizon and positive pitches aiming below the horizon.

Terms we will use in this paper frequently will be pitch, yaw, and force. Pitch and yaw refer to angles of direction of the player. Yaw refers to direction in the XY plane (i.e. facing north, south, east, west). Pitch refers to direction in the XY-Z plane (i.e. how far up or down the player is looking). Force refers to the length of time the bow is drawn (in seconds) with $0 \leq \text{force} \leq 1$. This is then used to calculate initial projectile velocity as $v_o = 2f + f^2$.

II. METHODS

Explain the simplification of the problem from a 3D problem into a 2D problem by turning the XZ plan into radial coordinates and a XZ - Y plane to model the projectile motion. Explain how pitch, yaw and force play a part in projectile motion.

We simplified the problem from a 3D problem to a 2D problem by eliminating the need for calculating yaw. We decided this was an unnecessary component to learn since the yaw necessary to hit the target is simply the direction of the target and is simple to compute. With this we are left with a 2D problem with unknown parameters pitch and force.

Explain the approach to getting the data. (Using project malmo, teleporting around and determining random targets) How we wanted to make inputs similar to information a singular person will have standing in this world would have.

In curating a data set we used project Malmo to facilitate communication with Minecraft. Using Malmo, we created a script that generated a hilly terrain then repeatedly, randomly placed the target and the player somewhere in the

generated terrain and brute forced a combination of pitch and force that managed to hit the target. This was done by starting with a pitch of -89 (looking nearly straight up) and testing various values of force. If none hit, the angle was slightly reduced and tried again. Once a pair of pitch/force was found to hit the target, those values were recorded in addition to the environment directly between the player and the target. This way, our initial dataset included the pitch/force, the positions of the target and the player, and the heights and distances of obstacles between the player and target.

The different features we considered in using in our dataset. (Elaborate and why we would want a feature, Such as, the relative position of the target from the player. The absolute position of the target.) The features we decided to use in our dataset. (We decided to not use the absolute position of the target from the player as that information is encoded in the relative position of the target. Using all relative positions allows the model to learn Minecraft's measurement of distance [Block] and maybe learn Minecraft's measurement of time [Tick].)

We decided to use coordinates relative to the player to make the model learn from the perspective of the player. We also thought it would make the model more accurate by pre-computing the relative coordinates rather than have the model learn them.

We also decided to only use the height and distance of the tallest obstacle between the target and the player. Situations in which the arrow clears the tallest obstacle but gets stuck on another is exceedingly rare and including those cases in our exploratory models drastically reduced accuracy.

Show and explain graphs of the plotted data. 'Hittable' vs 'Non-Hittable' Linear shape of fully drawn shots that hit. Explain why we split the data into 'Hittable' and 'Non-Hittable' (See if we can predict this) Explain why we split the data into 'Fully draw bows' and 'Varying draw strength'

I'll leave this part to you since you did the

hittable/non-hittable.

State we created a simulator for arrows shot in Minecraft for the below and what the simulator actually does.

I'll leave this to you as well. You should know however that I ended up using my simulator for the arrows for the results section. I found that mine was consistently accurate and created a results metric that was more accurate to what was happening and looked better as well.

Explain how we determined the error/constraint metrics. If the model got the same pitch/yaw/force as the simulator (Terrible metric and why. The model could be providing similar, but not exact specs, that still hit the target.) Difference of the pitch/yaw/force of the model to the simulator. (Terrible metric and why. Huge outliers that might still hit the target.) Distance of arrow to center of the target. (Good metric and why.)

We used several types of error metrics before deciding on our final metric. At first we used square loss between target and predicted values. This method worked well for training models to predict pitch on max force, however there were still issues in that if there were no obstacles in the way a max force arrow could either be fired directly at the target or in a high arc.

Also, square loss didn't work well for multi-value prediction. Because of the nature of projectile motion, there are a whole family of solutions of pairs of force and pitch. Because of this, a model could have terrible training and test scores but the predicted values could still let the arrow hit the target.

We also tried a hit/not hit metric that trained on whether the predicted values would hit the target in the simulation or not. This worked reasonably well however the metric wasn't very informative of the final result. It only showed how many of the shots actually hit the target but for the shots that missed there was no information.

Our final error metric determined distance of the arrow from the center of the target. This

resulted in a great error metric since we could classify arrows with a small enough distance as "hit" and the rest as "non-hit". From there we could see whether the model is only good for predicting certain shots and terrible at others or whether the missed shots were still relatively close to the target.

Go through each method tested starting with linear systems. How we used ridge regression to find different parameters. Hypothesized that yaw was non-linear due to the use of tanh in it's equation. Hypothesized that with full force shots the underlying function is probably linear to an extent (remember drag!). Otherwise, it will probably be non-linear.

I'll let you organize this part. I'll throw up the ipython notebook with the new error metrics. I'll run them myself as well and see what I can throw into this section.

Talk about how we used an SVC to determine if a target is hittable or not. Talk about the hypothesis that the SVC will probably not do much given the prior graph on the matter (Hittable vs Non-hittable)

I don't know about this one.

Go through non-linear systems using Random Forest and why we believe these will perform better than the ridge regression solutions.

Again, I'll have them up.

Go through the neural network systems and explain why we're testing these.

III. RESULTS

I'll throw up graphs and tables onto the github and let you put them in where you like. In the mean-time I'll do some last-minute touch ups to the results data. The final data results are done, its mostly the other models and graphs that I'm working on.

How some parameters were assumed to be non-linear and provided terrible results with just linear systems. How adding non-linear kernels in Ridge regression helped but did not provide a complete/adequate solution.

Table 1: *Example table*

Name		
First name	Last Name	Grade
John	Doe	7.5
Richard	Miles	2

Speak about the results of the SVC to determine hitable and non-hitable.

Speak about the results of the Random Forest models.

Speak about the results of the NNs.

Speak about how the model found some solutions that were better than the simulator.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

$$e = mc^2 \quad (1)$$

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo.

Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

IV. DISCUSSION

We are pleased with the results and have automated the player using our models. First, use the SVC model to determine if a target is hitable or not. If it is hitable, use the NN to determine the pitch, yaw and force.

Speak about how we would suggest approaching similar circumstances where machine learning can aid in finding hidden physical functions.

Suggest future work. Such as determining projectile motion on different planets/video games.

i. Subsection One

A statement requiring citation [?]. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

ii. Subsection Two

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam

nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

REFERENCES

The two reddit users and links to their accounts (for the equations) Project malmo github page Rosenberg's Github because he taught the class (and is awesome) Raise fits in triumph due to completion of paper. Assortative pairing and life history strategy - a cross-cultural study. *Human Nature*, 20:317–330.