Estimating Caloric Expenditures of Hiking Routes from Elevation Profiles

Noah Lomas

Abstract:

In this paper, we build a model to estimate energy expenditure for a hiking route specified by a GPX file using a GPX recording as pace data and the hiker's weight. The resulting energy expenditure estimates were realistic for sustained-pace, walking-speed, single-day trips, but inaccurate for multi-day trips or trips with long break periods. A program called Calories.py was built to run the computation for this model.

Introduction

In recent years, hiking has exploded in popularity. The number of yearly hiking participants in the US has increased by 16 million in the past 5 years [1]. With the influx of new participants in outdoor recreation, there are many new hikers with little to no understanding of proper nutrition principles. A common dilemma for new hikers and backpackers is deciding how much food should be brought on a trip. There are many online "hiking calculators" that give misleading or simply wrong information about calorie burn. The goal of this paper is to create a model that can estimate a hiker's calorie expenditure based on their pace, weight, and an elevation profile of their route. This paper will be broken into several sections:

- Assumptions
- Model and Methods
- Results

Assumptions:

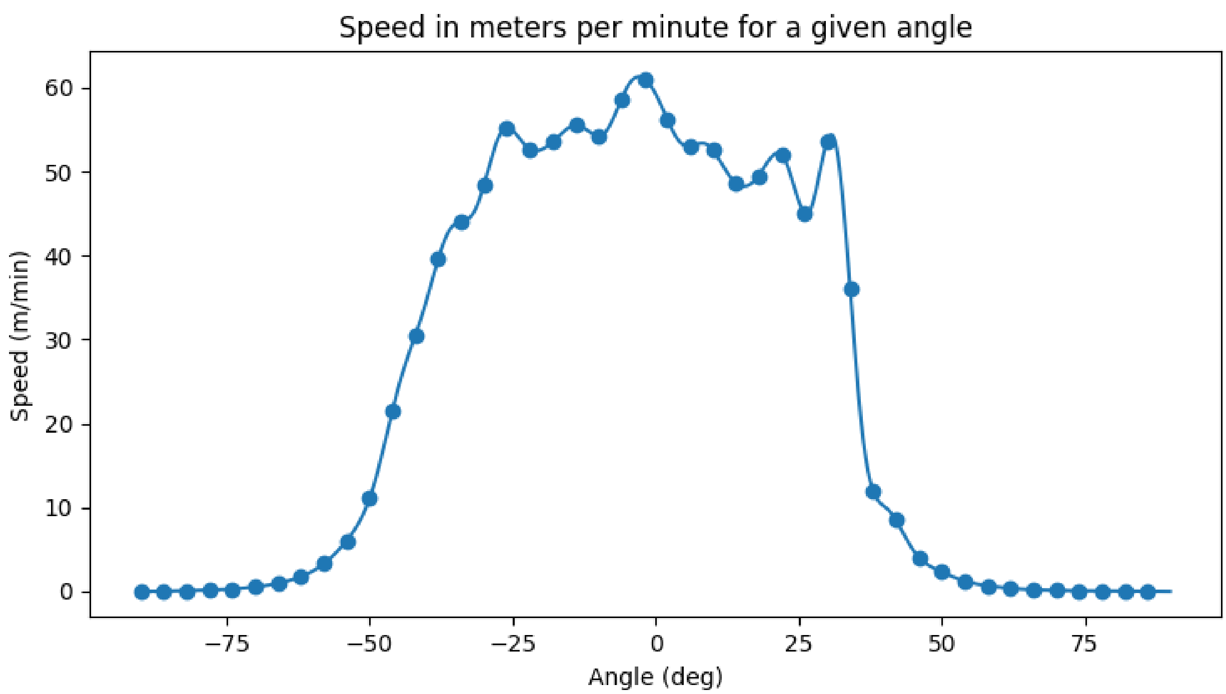
This paper attempts to simulate a hiker's walk along a trail. Below is a list of assumptions the model relies on:

1. The hiker's weight W is constant.
2. The position of the hiker, P, will vary as a function of time.
3. The steepness of the trial, G, will vary as a function of the hiker's position.
4. The speed of the hiker, S, will vary as a function of the steepness of the trail.
5. The oxygen consumption per unit mass of the hiker, $VO2$, will vary as a function of the speed of the hiker and the steepness of the trail (ACSM walking equations).
6. The energy expenditure of the hiker, EE, will vary is a function of their weight and oxygen consumption.
7. The difficulty of the terrain, the altitude, the weight of a hiker's backpack, and other external factors will not affect the hiker's energy expenditure.

The steepness-speed curve:

It's no secret that walking uphill is slower than walking on flat ground. But while we have a set of equations that quantify oxygen consumption, these equations depend on the steepness of the trail and the speed of the hiker, both of which are ultimately time dependent. This means we need some way to quantify how long a certain piece of trail will take. Enter: the steepness-speed curve. This curve is expressed as a function $S(m)$, where m is measured as a fractional grade. In general, the steeper the hill (the greater m is), the slower the hiker's velocity v.

Given that different hikers travel at a wide variety of paces, it makes more sense to cater a specialized curve for a given person. To generate this curve, we can use a "sample pace" GPX file. This is a GPX recording the hiker has already made (can be done using their phone or a relatively inexpensive GPS device) that will serve as a frame of reference for how quickly they walk on different grades. The Python function analyzeRecording takes a given GPX recording and outputs a steepness-speed curve as a list of points or csv file. It does this by grouping different steepness levels into a specified number of bins. Every steepness in the bin is measured for the speed the hiker was walking on that segment of trail. These steepnesses are then averaged to create a value representative of the entire bin. The speeds for some grades are not available, so to avoid zero values (which would essentially mean the hiker would get "stuck" at a certain grade) we assume any unknown bin is simply the average of the bins directly greater than and less than it. This has its limitations; future models may benefit from approximating these unknown values via data from other GPX recordings or clinical trials, but these sorts of approximations are beyond the scope of this paper.



The Steepness Function:

Using our steepness-speed curve from above, we can now calculate steepness at a given time. We'll call our steepness function $G(t)$ and make the units of its output a fractional grade. Here's how this function's output is calculated for a given t:

Imagine the GPX file can be described as a series of n points representing a location in space. These points are stored in an ordered list:

$$P = [p_1, p_2, p_3, \dots, p_n]$$

For every line between two successive points p_k and p_{k+1} (each is a vector with latitude, longitude, and height) in our GPX file, we will "simulate" a hiker walking along it. Their speed will be determined by the slope of the trail between these two points. This speed can in turn be used to calculate the time elapsed (E) for a hiker to walk between p_k and p_{k+1}. We will also keep track of a total time elapsed, E_t, which will start at 0.

$$E = \frac{\sqrt{d(p_k, p_{k+1})^2 + h(p_k, p_{k+1})^2}}{S\left(\frac{h(p_k, p_{k+1})}{d(p_k, p_{k+1})}\right)}$$

In this equation, d is a library function to calculate the lateral distance between two latitude-longitude coordinates. h is another to calculate the vertical distance. Combined with the Pythagorean Theorem, they describe the total distance between points a and b. Now that we know how much time it takes for the hiker to get to point p_{k+1}, we can set $k = k + 1$, $E_t = E_t + E$, and repeat this step. During each step, we can extract an xy coordinate:

$$\left(t, S\left(\frac{h(p_k, p_{k+1})}{d(p_k, p_{k+1})}\right)\right)$$

Which will serve as the value for $G(t)$.

Calculating V02 at a given time:

The ACSM walking equation is defined as follows:

$$VO2 = (0.1 \cdot S_w) + (S_w \cdot G_w \cdot 1.8)$$

Where S_w is speed in meters per second and G_w is percent grade [3]. The output, VO_2, is in units of $(ml) \cdot (kg)^{-1} \cdot (\text{min})^{-1}$. Using our steepness function and steepness-speed curve, we can calculate VO_2 at a given time t:

$$VO2_{pos}(t) = (0.1 \cdot S\big(G(t)\big) + (S(G(t)) \cdot G(t) \cdot 1.8)$$

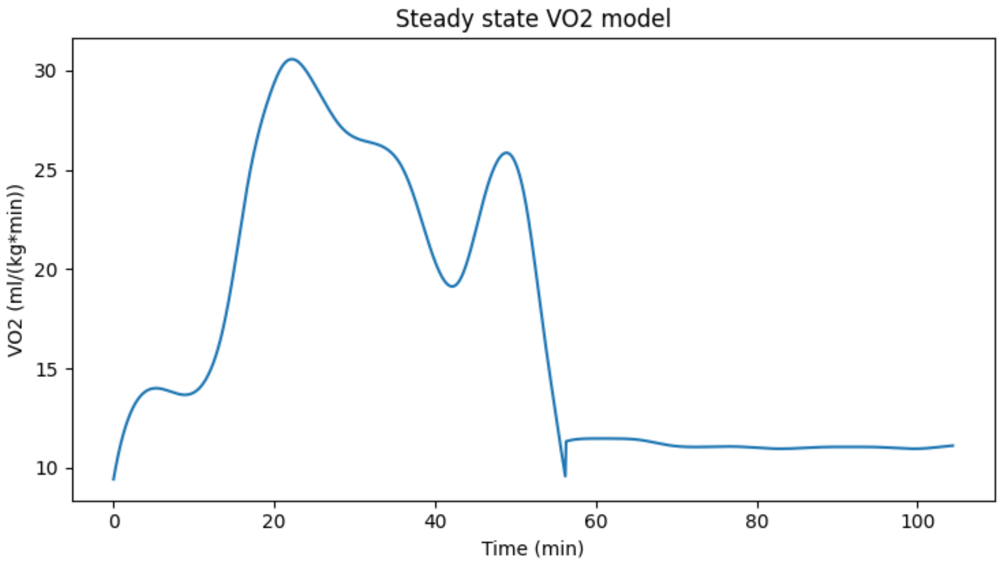
Note that this equation is only satisfactory for positive grades. For negative grades, we use a modification of VO_2 on flat ground [6]:

$$VO2_{neg}(t) = 0.73 \cdot (0.1 \cdot S\big(G(t)\big)$$

Together these compose a piecewise function for steady state VO_2:

$$VO2(t) = \begin{cases} VO2_{pos}(t) \text{ if } G(t) > 0 \\ VO2_{neg}(t) \text{ if } G(t) < 0 \end{cases}$$

This function is considered to be accurate at higher grades and lower speeds and is thus ideal for hiking [2].



Non-steady state exercise:

There is an issue with our method of calculating VO_2. The ACSM walking equations were designed for steady state exercise. In short, this refers to exercise of a constant intensity. Hiking is an exercise of varying degrees of intensity (moment-to-moment energy expenditure is dependent on speed and grade) so the ACSM walking equation alone is not an appropriate model. To improve our approximation of EE, we will use logistic functions that model the transitions between periods of steady state exercise. These functions draw heavily from a 2015 model used to calculate EE from heart rate and accelerometer data [7]. While our model uses neither, the functions in this paper were just functions of time, so the curves fit for energy expenditure differences between slow walking and fast walking will suffice for our model. We use a simplified version of their logistic function:

$$VO2_{NS}(t) = \frac{\theta_1}{1 + e^{-k(t-a)}} + VO2_{SS}(t_E)$$

Where θ_1 is the difference in steady state VO_2 before and after the transition, k is the steepness of the function, and a is the time offset.

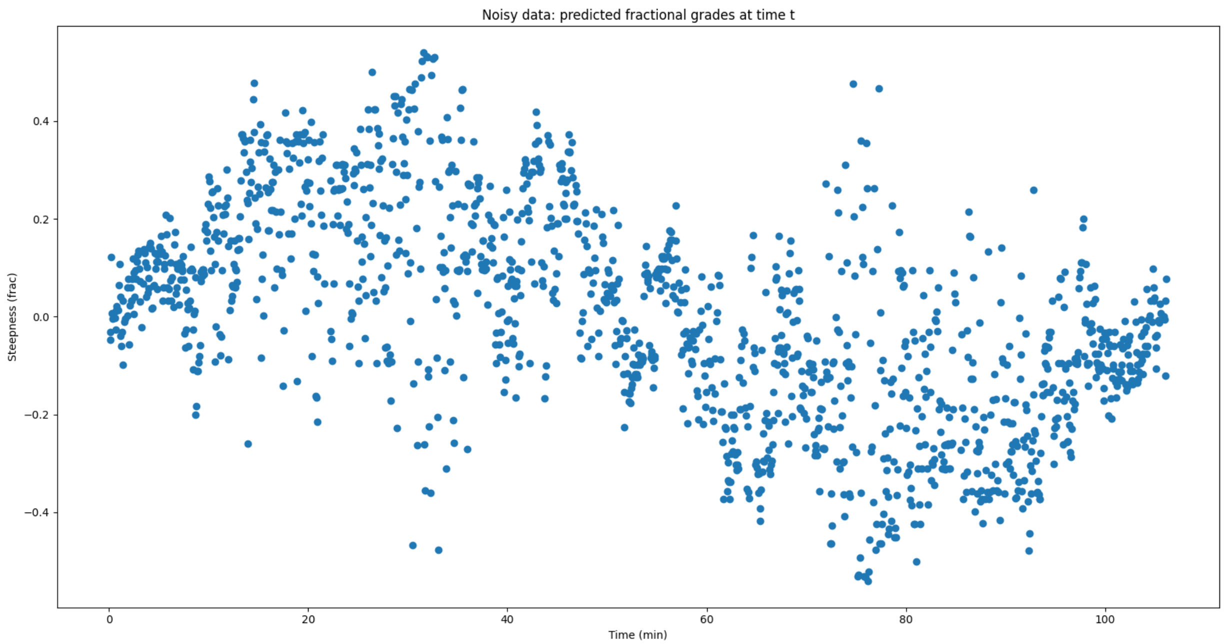
Note the absence of the original variable names present in the original model; these variables were for curve fitting that was optimized using data collected in the study. The values for these variables were not shared in the paper, so we'll have to make some simplifying assumptions to calculate them.

Assumption: $a = \left(\frac{t_B + t_E}{2}\right)$

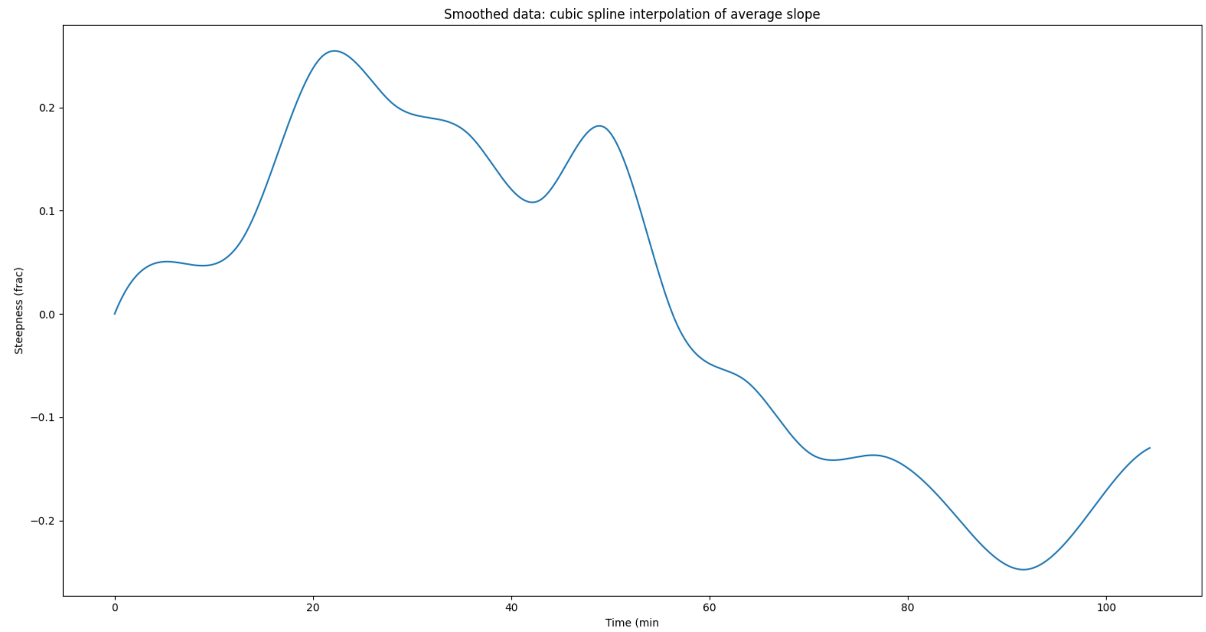
Where t_B and t_E are the times at when the transition function begins and ends. This assumption essentially states that the hiker's VO_2 at the time halfway between the two steady states is an average of the VO_2 at the two steady states. This will create a symmetry when we integrate this function that will simplify the problem considerably.

Detecting non-steady state exercise:

On top of calculating intervals of steady and non-steady state exercise, we need to calculate where these intervals begin and end. To do this, we'll again borrow from the methodology of [7]. But there's a big issue with this: detection of non-steady state exercise periods depends on slope. Slope as derived from GPX data is very erratic. If we use the methodology of this paper, the hiker will virtually always be in non-steady state exercise!



In the process of writing this paper, I tried several methods of smoothing the grade data to group exercise states more easily, but the only realistic solution is to greatly reduce the resolution of the grade data. For a given interval of grade data, we calculate slope over the entire interval instead of the slope at every point. This results in a smoother curve albeit with lower resolution. A series of cubic splines interpolate these points to give a broader idea of the trail's difficulty over time. In real-world scenarios, this is likely more realistic than a constantly changing slope. Most hikes consist of flat sections, climbs, downhills, and these sections are generally grouped together in large, 10-to-20-minute chunks.



These splines will also generate a less erratic VO_2 function. With this function, we can finally determine when steady state exercise periods will start and end.

A widely well-regarded threshold for non-steady state exercise is a 10% difference over a 4-minute period [4]. We'll create a function that measures the percent change in the function over a 4-minute window and calculate the intervals where the value is below 10%. The values inside these intervals will be used to calculate steady state VO_2, and the values outside will be used to calculate non-steady state VO_2.

Calculating oxygen expenditure per kilogram for the steady state model:

Integrating our VO_2 function with respect to time will yield the total oxygen intake in milliliters per kilogram over the entirety of the hike:

$$O_{SSm} = \int_0^{T_f} VO2(t)dt$$

This value can then be combined with the total non-steady state oxygen intake to calculate total caloric expenditure.

Integrating the transition function:

The trick to simplifying this problem is to shift the function and bounds of integration:

$$\int_{t_B}^{t_E} VO2_{NS}(t)dt = \int_{t_B+a}^{t_E+a} VO2_{NS}(t+a)dt = \int_{t_B+a}^{t_E+a} \frac{\theta_1}{1+e^{-k(t+a-a)}} + VO2_{SS}(t_B)dt$$

$$\theta_1 \cdot \int_{t_B+\left(\frac{t_B+t_E}{2}\right)}^{t_E+\left(\frac{t_B+t_E}{2}\right)} \frac{1}{1+e^{-k(t)}}dt = \theta_1 \cdot \int_{t_E-(t_B+t_E)}^{t_B+(t_B+t_E)} \frac{1}{1+e^{-k(t)}}dt = \frac{\theta_1}{2} \cdot (t_E - t_B)$$

In the end, our resulting function is simply $(t_E - t_B) \cdot (\frac{\theta_1}{2} + VO2_{SS}(t_B))$.

Calculating total energy for the non-steady state model:

Adding everything together, our final model for oxygen expenditure per kilogram looks something like this:

$$O_m = \sum_{k=0}^{n} (I_{k+1}^B - I_k^E) \cdot \left(\frac{VO2_{SS}(I_{k+1}^B) - VO2_{SS}(I_k^E)}{2} + VO2_{SS}(I_k^E) \right) + \int_{I_k^B}^{I_k^E} VO2(t)dt$$

Where I_k^B is the beginning bound of the kth interval and I_k^E is the ending bound of the kth interval.

To convert O_m into Calories, we'll multiply by the weight of the hiker to get total oxygen intake, then by the average Calories burned per liter of oxygen [5].

$$EE = W \cdot O_{SSm} \cdot 5$$

This should yield the total number of Calories burned over the entire hike.

Results:

The model was run on two "sample" GPX recording files to generate steepness-speed curves, which were then subsequently ran on a handful of GPX route files. The figures for these routes are presented in the table below, along with some basic information about the trail.

Route Name	**Total Calorie Burn (kCal)**	**Total Distance (mi)**	**Total Elevation Gain (ft)**
SnowslideLakeTrail.gpx	812.19	3.5	1,328
GoatLake.gpx	1,893.06	8.1	1,768
CactusCloud.gpx	5,221.23	21.2	10,764
GreatRangeTraverse.gpx	3,581.60	19.5	8,917
JohnMuirTrail.gpx	28,074.70	210.1	46,030

The model projected realistic figures for energy expenditure, with the exception of the John Muir Trail. The John Muir Trail is a 210-mile trail in California that takes most hikers 2 to 3 weeks to complete at 2,500 to 3,000 Calories a day. With the diet suggested by this model, you'd be consuming a mere 1,300 Calories while hiking 10+ miles a day! The reason for this discrepancy is that this model does not account for the energy expended while resting. On multi-day hiking trips, a great deal of time is spent sleeping and taking breaks. During this time, Calories are still being burned. One possibility for future models is to factor in the time used and Calories burned for rest periods.

One interesting facet of the model is the low effect hiking speed along has on VO_2. On downhill sections, where grade is considered “flat” based on aforementioned approximations, VO_2 fluctuates very little. When large changes in grade are involved, VO_2 fluctuates wildly.

It’s difficult to say exactly how accurate this model is objectively, as there’s not much hard data to compare it to. Based on a few tests and my own experience packing adequate (and inadequate) amounts of food for hiking trips, I’d say I’d trust this model to give a decent “ballpark” figure for energy expenditure on single-day, walking-speed trips.

Citations:

[1] Stastista Research Department. “U.S. Americans Who Went Hiking 2021 | Statista.” *Statista*, www.statista.com/statistics/191240/participants-in-hiking-in-the-us-since-2006. Accessed 5 Dec. 2022. [2] Dalleck, Lance C et al. “The accuracy of the American College of Sports Medicine metabolic equation for walking at altitude and higher-grade conditions.” *Journal of strength and conditioning research* vol. 19,3 (2005): 534-7. doi:10.1519/1533-4287(2005)19[534:TAOTAC]2.0.CO;2

[3] Glass, S. & Dwyer, G.. (2012). ACSM's Metabolic Calculations Handbook.

[4] Reeves, Marina M et al. “Reducing the time period of steady state does not affect the accuracy of energy expenditure measurements by indirect calorimetry.” *Journal of applied physiology (Bethesda, Md. : 1985)* vol. 97,1 (2004): 130-4. doi:10.1152/japplphysiol.01212.2003

[5] Janot, Jeffrey M. "Calculating caloric expenditure." *IDEA Fitness Journal* 2.6 (2005): 32-33.

[6] Weyand PG, Ludlow LW, Nollkamper JJ, Buller MJ. Real-world walking economy: can laboratory equations predict field energy expenditure? J Appl Physiol (1985). 2021 Oct 1;131(4):1272-1285. doi: 10.1152/japplphysiol.00121.2021. Epub 2021 Aug 19. PMID: 34410843; PMCID: PMC8560389.

[7] Altini M, Penders J, Amft O. Estimating Oxygen Uptake During Nonsteady-State Activities and Transitions Using Wearable Sensors. IEEE J Biomed Health Inform. 2016 Mar;20(2):469-75. doi: 10.1109/JBHI.2015.2390493. Epub 2015 Jan 12. PMID: 25594986.