Oxygen, crampons, data

Can statistics help us climb Everest?

Climbing mountains is a risky business, and mountaineers disagree on the best way to do it. **Moinak Bhaduri** explains how a statistical approach can give your expedition the best chance of success

It doesn't take a professional alpinist to grasp that a mountain's height is a dodgy indicator of climbing woes.

If you or a commercial company were about to organise an Everest expedition and were unsure of the "right" way to juggle a ton of crucial decisions, what may resonate more would be some data-based evidence that chalks out the best general plan. This plan would consider the route, team size and diversity, oxygen usage, time to acclimatise and so on, and also offer guided specifics along with a rule that, once you plug in your choices, spits out your probability of success.

Let's construct some such models, that is, work out good prediction formulae – using a random portion of the available data, called the *training set*. We'll then check their accuracy using features from the other portion – the *test set* – unseen data that was not used to construct them, and all through a balanced split, that is, the proportion of successful expeditions remaining unaltered over both the training and test sets.

The raw data (i.e., the first and second columns in Table 1) are from the Himalayan Database maintained by Elizabeth Hawley, Richard Salisbury and others, covering 2,214 Everest expeditions since 1921 (www.himalayandatabase.com). The 1996 climbing season was the deadliest, claiming even renowned climbers Scott Fischer and Rob Hall, among many others. The details of Fischer's expedition are shown in the sixth column of Table 1.

Imagine we are planning an expedition roughly similar to Fischer's – using the same South Col–SE Ridge route (the one used by most Everest expeditions since 1950), facing a similar amount of crowding, implementing

The 1996 climbing season was the deadliest, claiming even renowned climbers Scott Fischer and Rob Hall

Moinak Bhaduri develops change-detection algorithms for point processes and is an assistant professor with the Department of Mathematical Sciences, Bentley University. He also heads the editorial board of the NextGen column of the *New England Journal of Statistics in Data Science*.

Table 1: We list the predictors, the responses (binary "success" and numerical "maximum height attained") and the tree-based models connecting them. Factors which trigger the greatest response homogeneity in the subsequent nodes (detailed below) receive the highest "importance score". The most crucial factor is shown in red, the next in green, the third in blue, the fourth in orange, and the fifth in purple. The gradients summarise the way in which changes in these factors impact the log-odds of success. The (+,0,–) on "N.Teams", for instance, indicates that the log-odds improve initially with more teams on the mountain, then become stagnant and eventually fall, evidenced also in Figure 3(d) below.

Features	Description	Summary (median, mean, SD) for quantitatives	Importance score (BaggedTree, RForest, TreeNets)	Partial dependence plot gradient	1996 Expedition (Scott Fischer)	A future expedition planned for 2024
Season	Spring (1), summer, (2) autumn (3), winter (4)	Four categories	(9.3, 10.2 ,3.1)	(–, +)	1	1
S.Days	No. of days to reach summit/ high point	(35,32.3,16.3)	(67.1,12.3,3.8)	(+)	33	65
T.Days	Total no. of days	(39,33.9,19.8)	(41.5,4.4 ,4.07)	(–)	39	70
Camps	Number of high camps above base camp	(3,2.9,1.5)	(75.1,26.5 ,6.4)	(+,0)	4	5
Rope	Amount of fixed rope (in metres)	(0,137,677.9)	(2.67,0.3 ,0.47)	(–,+,0)	0	0
T.Mems	No. of members	(5,7.1,7.5)	(30.3, 1.3,6.14)	(+,0)	12	15
T.Hired	No. of hired personnel above base camp	(3,6.0,8.9)	(33.1, 6.0 ,3.3)	(+,0)	10	0
Route	The route taken	148 options	(58.6,6.8,48.4)	–	S Col–SE Ridge	S Col–SE Ridge
O2Climb	Oxygen used while climbing	Yes or No	(100, 100, 100)	–	Yes	No
O2Sleep	Oxygen used while sleeping	Yes or No	(76.9,34.5 ,8.7)	–	Yes	No
O2Medical	Oxygen used for medical emergencies	Yes or No	(0.6, 0.2 ,0.32)	–	No	No
Number of teams on Everest that season	N.Teams, possibly correlated with Year Index	(77,62.8,33.4)	(45.2,11.9,9.6)	(+,0,–)	30	32
Diversity	No. of different nationalities	(1,2.1,1.9)	(9.95, 0.4,0.36)	(+,0)	3	4
N.Routes	Total number of routes taken	(1,1.1,0.22)	(0.00, 0.0,0.00)	(+,0)	1	2
Year Index (start: 1921)	Possibly correlated with N.Teams	(87,85.2,12.1)	(43.7,8.8 ,6.5)	(+)	76	103
Success	Whether at least one member reached the summit	Yes (1) or No (0)			Yes (1)	?
Max Height	Highest altitude reached				8,849 m	?
Estimated probabilities		Success probabilities from (BaggedTree, RForest, TreeNets)			(0.76,0.84,0.89)	(0.16,0.17,0.13)
Estimated max height		Predicted height (in 100s of metres)			(91,91,93)	(67,64,69)

similar climbing strategies (the last column in Table 1). Can statistical models predict whether we too will reach the summit? If the chances look grim, what factors can we tweak to better our odds? Let's find out.

Let's acclimatise!

Climbing Mount Everest is no mean feat. And part of the standard strategy is to acclimatise properly by practising smaller climbs – "climb high, sleep low", so the saying goes – before the real push begins.

Building predictive models is not so different. Statisticians have a habit of building several quick, simple rules initially to get a rough view of the data terrain, without straining their modelling limbs. These go by different names: sometimes "naive models", sometimes "benchmarks". Like forecasting tomorrow's closing stock price by today's, or estimating an expedition's success probability just by the current overall success rate (it's 0.62 here, called the "no information rate"). Faced with a classification task – saying whether an expedition will succeed – we are hardwired to deploy the logistic model. Let's quickly climb this small peak, and one more, so we know what can be achieved pretty cheaply.

Given an expedition, we erect a tower at the right location on the "floor", measure the height at which this tower cuts the logistic probability surface along the "wall" (Figure 1(a) and "The recap box"). If this height exceeds a threshold of, say, 0.5, we will say the expedition will be successful. Such an agreement, on the test set, gives an accuracy of 0.69 (see Table 2). The logistic model comes with strings attached, however. If *all* the failed attempts line up with small values of a predictor *and* all the successful ones go with big values (a problem called separability) the estimates 0.77 and 0.79 become unstable. Furthermore, the linearity of form in equation (1) may be questionable. What if, no matter how large we make that equation, or how

iStock.com/sihasakprachum

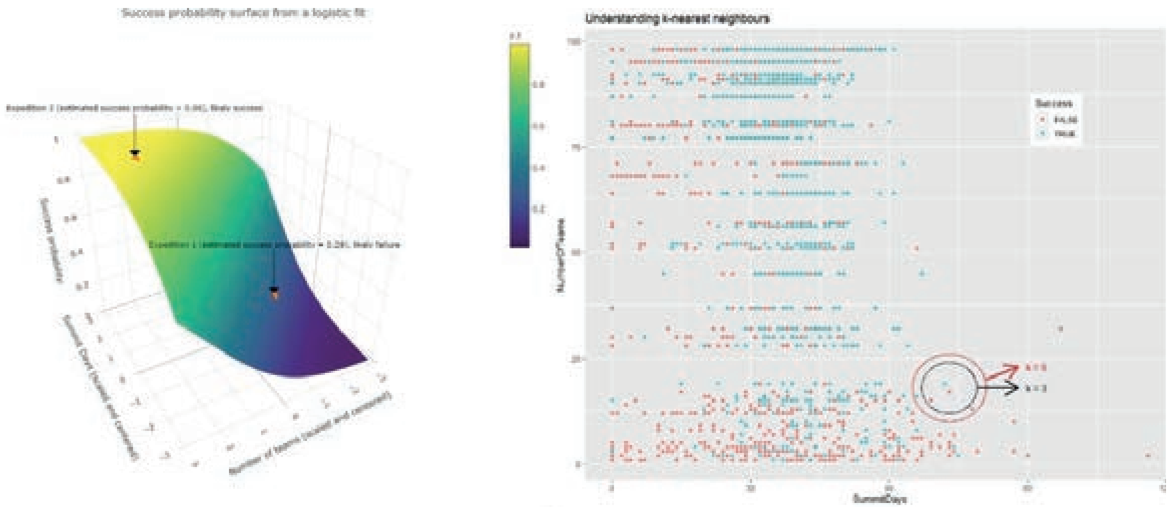

Figure 1: Acclimatisation (understanding simple classifiers). (a) Model (2) (see “The recap box”) graphed. How an expedition’s success can be “measured off” the logistic surface. Two trial expeditions are shown as orange blobs. (b) Failure verdicts for an expedition falling at the centre, from the KNN approach with three and five neighbours. Two out of three expeditions failed in the smaller neighbourhood, while three out of five failed in the larger. The majority vote, therefore, is “failure” in either case. The right neighbourhood size is chosen through cross-validation.

When it comes to our Everest expedition, we find the use of oxygen to be the most crucial decider

The recap box

To predict a binary outcome *Y*, recall how the logistic serves as the standard initial go-to. It models the log of the odds as a linear function of the predictors. For us, with 0 representing failure and 1 success, this becomes

$$\log\left(\frac{P(Y=1)}{P(Y=0)}\right) = 0.55 + 0.77X_1 + 0.79X_2, \tag{1}$$

where X_1 represents the number of days the team took for its summit push and X_2 the number of other teams on the mountain that season. We can express the success probability as

$$P(Y=1) = \frac{1}{1+\exp(-0.55-0.77x_1-0.79X_2}, \tag{2}$$

a surface shown in Figure 1(a).

wisely we choose the predictor variables, the way we are combining them – through adding – is just not right?

The *K*-nearest neighbour (KNN) approach rids us of some of these headaches (see Figure 1(b)). Armed with the particulars of a new expedition, the method scans history to locate comparable expeditions – its neighbours or friends, so to speak – and assigns to this new expedition a fate that greeted *most* of its friends. How many neighbours should we examine? That is, what should *K* be? We check different trial values and choose the one that maximises our accuracy within the training set. For us, *K* = 7, using cross-validation (10-fold, 5-run). The accuracy (in relation to the logistic) has gone up to 0.73.

Sadly, KNN is plagued by its own different set of problems. If some features are non-numerical (like the route they took, the season) defining a distance becomes tricky. Additionally, if we have lots of features/explanatory properties (we now have 15), any one dot in the scatter-cloud in Figure 1(b) is almost guaranteed to be lonely. The difference between a neighbour and a non-neighbour becomes minimal; it gets tough to distinguish a friend from an enemy – a trouble aptly dubbed “the curse of dimensionality”.

The real climb begins: some tree-based offerings

Trees do not grow on the slopes of Everest – not, in fact, anywhere above 4,800 feet. Within the modelling landscape, however, we do not have a “tree-line”. Classification trees, described below, are decision rules that flourish where others – like the logistic and the KNN – falter. They are shown in Figure 2 and are made to reduce the entropy – the chaos – present in our data. Imagine trying (without looking, of course) to say whether a creature is a squirrel or a human by asking just one question. The presence or absence of a tail would be a great classifier. All those with a tail will be placed in one category, all those without in another. The tail–squirrel connection is almost embarrassingly automatic, making us leak information perfectly. Initially, if we had 100 creatures – 50 squirrels and 50 humans – the chaos and our confusion are at their greatest. Following the split, within each subgroup we have creatures only of one kind – the chaos and confusion have disappeared, that is, we haven’t misclassified anybody. Make the decider, or the connection, any less automatic, by asking,

Table 2: The models' performance on the test set. No information rate (NIR) = 0.6178; 10-fold, 5-run cross-validation implemented to estimate the tuning parameters. High accuracies achieved despite the unavailability of crucial uncontrollable predictors like the weather (equivalent information may, however, be contained in "Total Days", for instance).

Metric	Description	Logistic	KNN	BaggedTree	RForest	TreeNets
Classification accuracy	Rate at which expeditions get correctly classified: the higher, the better	0.692	0.7301	0.8877	0.8822	0.8895
95% confidence interval	A measure of how unsure we are with these point estimates: the narrower, the better	(0.65,0.73)	(0.69,0.77)	(0.85, 0.91)	(0.85, 0.91)	(0.86,0.91)
p-value(Acc > NIR)	Have we done significantly better than the NIR classifier? The smaller, the better	0.00016	1.75×10^{-8}	$< 2.2 \times 10^{-16}$	$< 2.2 \times 10^{-16}$	$< 2.2 \times 10^{-16}$
Classification sensitivity	Rate at which successful expeditions get correctly classified	0.8475	0.8475	0.9443	0.9589	0.9355
Classification specificity	Rate at which unsuccessful expeditions get correctly classified	0.4408	0.5403	0.7962	0.7583	0.8152

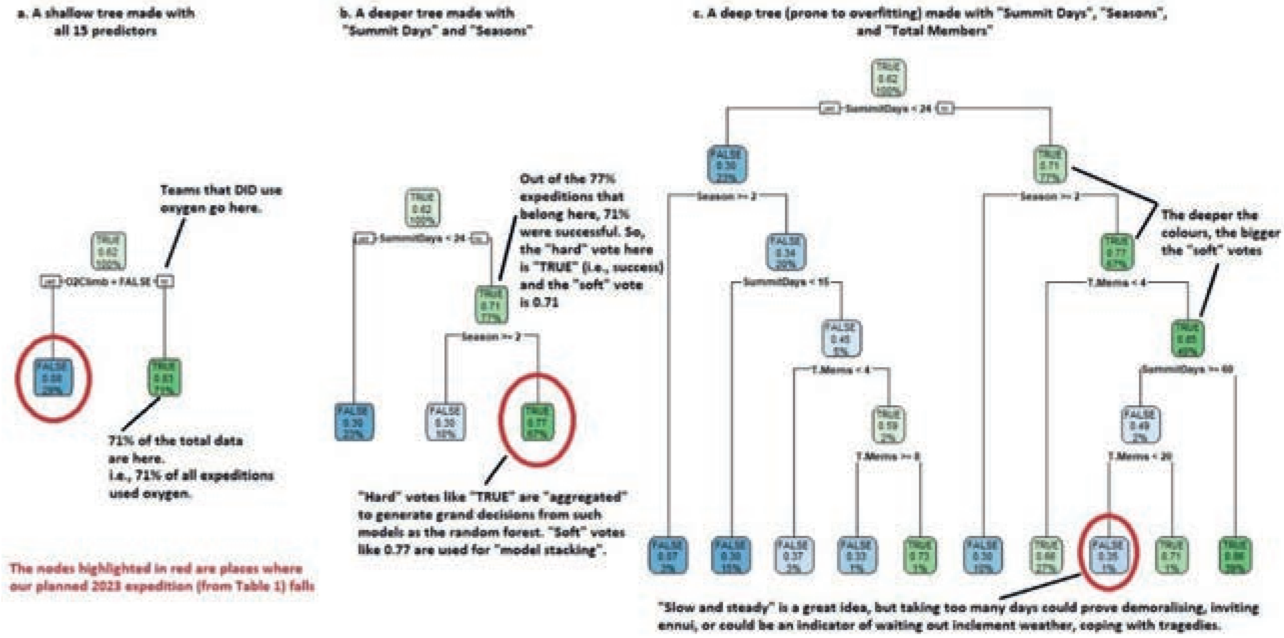
a. A shallow tree made with all 15 predictors
b. A deeper tree made with "Summit Days" and "Seasons"
c. A deep tree (prone to overfitting) made with "Summit Days", "Seasons", and "Total Members"
Teams that DID use oxygen go here.
O2Climb = FALSE
yes
no
71% of the total data are here. i.e., 71% of all expeditions used oxygen.
SummitDays < 24
Season >= 2
Out of the 77% expeditions that belong here, 71% were successful. So, the "hard" vote here is "TRUE" (i.e., success) and the "soft" vote is 0.71
"Hard" votes like "TRUE" are "aggregated" to generate grand decisions from such models as the random forest. "Soft" votes like 0.77 are used for "model stacking".
The nodes highlighted in red are places where our planned 2023 expedition (from Table 1) falls
SummitDays < 15
T.Mems < 4
T.Mems >= 8
SummitDays >= 60
T.Mems < 20
The deeper the colours, the bigger the "soft" votes
"Slow and steady" is a great idea, but taking too many days could prove demoralising, inviting ennui, or could be an indicator of waiting out inclement weather, coping with tragedies.
TRUE
FALSE

Figure 2: Ways in which trees work. The splits at any stage are controlled by features (and their values) that make the observations in subsequent nodes most homogeneous.

say, "Do you like walnuts?", and the split will not be quite so neat. Each subsequent subgroup may contain creatures of both types (yes, squirrels like walnuts but some humans do too). So, features (like a tail) and their values/categories (like yes/no) that do the separation most efficiently creep to the top of these trees in the hope of doing the heaviest lifting. Any impurities that persist will be dealt with later down the tree with other features/values – those that are less efficient.

When it comes to our Everest expedition, we find the use of oxygen to be the most crucial decider. So, to predict an expedition's outcome using *one tree*, start at the top, take the right twists and turns, depending on its properties, and reach a terminal node. The verdict is whichever outcome is more common in that node.

These trees, to generate perfect predictions on the training set, can go quite deep, with lots of splits (i.e., branches). The reliability of their predictions suffers, that is, their variances can be big (recap, for instance, overfitting from regression). Slight changes in the data could lead to hugely different verdicts. A common remedy is to "bag" (i.e., bootstrap-aggregate) a lot of these deep, isolated trees, creating a "bagged tree". At the bootstrap stage, resamples, say 500 of them, $D_1, D_2, \ldots, D_{500}$, are drawn from the original set of 2,214 expeditions and 500 deep trees, $T_1, T_2, \ldots, T_{500}$, are made: T_i using data D_i.

We can funnel your expedition, with all its predictors, through each one of these 500 trees and if, say, 400 of those 500 decisions are "success", we can say the expedition is likely to succeed. This is the aggregation stage. We go with the majority vote. Borrowing such strength from the crowd works best if the members of the crowd are unconnected to each other – if each member has an

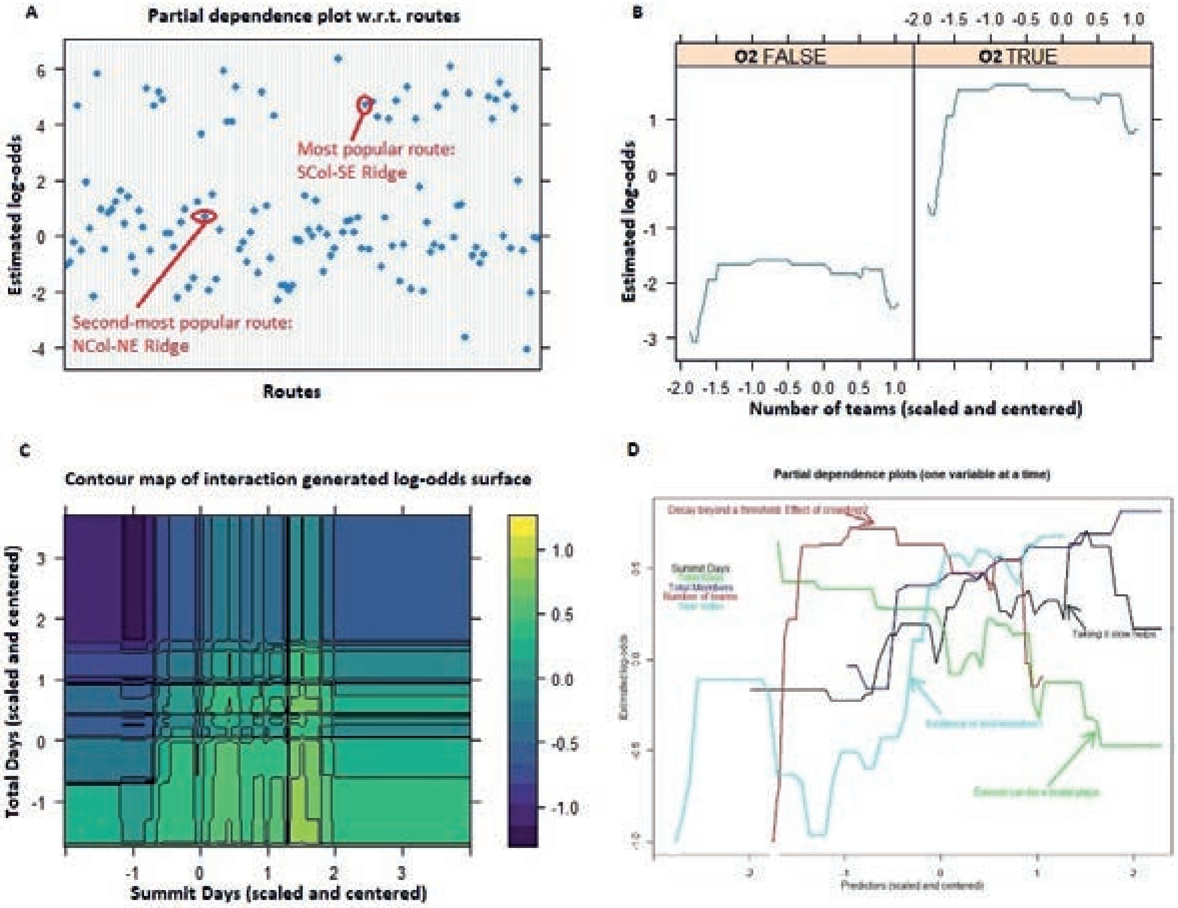

Figure 3: A sampling of interesting dependence plots (more on the dashboard, see footnote). Numerical variables are centred and scaled to ensure each may exert a similar influence on the odds, regardless of its scale.

independent voice, that's when the variance reduction happens. To ensure these 500 trees are unconnected, a trick is to offer not all, but only a randomly chosen handful of predictors at each split, leading to a "random forest". Another option, pioneered by Friedman,[1] is a "TreeNet", where several shallow trees (known as "weak learners") are made sequentially, each fixing the mistakes made by the immediately previous one. Table 1 and Figure 2 show how the fate of a future expedition (ours) or a past one (Fischer's) may, this way, be decided or debated in retrospect. Notice how, funnelled through the shallowest and deepest trees in Figure 2, our planned 2024 expedition ends up in nodes where "failure" is more common (the circled nodes). These decisions – those that pick the more popular category – are called "hard" votes, as opposed to the "soft" votes shown, which quantify the margin of victory. If these three trees are parts of a random forest, "FALSE" (i.e. "failure") becomes the grim final verdict, following the majority (two out of three) of the hard votes, that is, implementing the aggregation. What went against us? Although our expedition was similar to Fischer's – and that team could reach the top – we differ crucially on our attitudes towards supplemental oxygen: while they used it, we plan not to.

The importance scores on Table 1 reveal which predictors consistently offer the most help in reducing entropy. Those with higher scores act like our "tail", silencing the chaos across most of the component trees. The use of oxygen, the route taken, the number of teams on the mountain, and the number of days taken to reach the summit emerge as the consistent critical deciders from these tree-based models. These scores, however, are blind to the direction of correlation – a problem remedied by the partial dependence plots. The red curve on Figure 3(d), for instance, shows that (assuming other factors are controlled) an increase in the number of teams could gradually elevate the estimated log-odds of success by 1.5. This may be due to the removal of isolation, fostering cooperation and camaraderie – but the rise, in keeping with mountaineers' beliefs, is short-lived, giving way to a gradual fall once several teams begin crowding.

You may even query factors that are less quantifiable such as desire, ambition and tenacity

iStock.com/marozuc

Interaction plots like Figure 3(c) document how these factors mesh and clash. Figure 3(a) shows that among the 148 possible routes (each a blue dot), there are a distinct few (such as the South Col–SE Ridge) that substantially elevate the odds of success. Figure 3(b) reveals how the use of oxygen while climbing can correspond to an almost three-unit improvement in the estimated log-odds regardless of the crowding. Purists like Italian mountaineer Reinhold Messner insist on not using supplemental oxygen. So if you want to not just climb, but meet Messner's standards, expect a battle three times as tough. Or how about a climb without oxygen on a treacherous route? A different interaction plot will be appropriate then (the Shiny dashboard will help, see footnote).

Figure 3(c) locates a specific planning window that could offer the greatest chance of success. It suggests (through the fainter shades at the bottom middle) taking our time while climbing but getting the entire expedition over quickly. Figure 3(d) shows the dependence of the log-odds on one quantitative variable, holding the others fixed. Notice, for instance, how despite an initial rise, the log-odds fall beyond a crowding threshold of 46 teams (–0.5 normalized "number of teams"). The literature is littered with tragic examples of climbers who froze amidst jams on the Hillary step or of those who fell to their deaths while trying to unclip and reclip themselves while overtaking slower climbers.

This is a neat start. Still, climbing gear improves over time. You may even query other factors such as a climber's sex, fitness, climbing experience, or those that are less quantifiable such as desire, ambition and tenacity. Models built using these details may offer fresher probabilities. Models built here may serve as benchmarks (just as the logistics and the KNNs were for us) against which these newer ones may shine.

What about climbing other peaks beside Everest? Or calculating, – for personal records or securing future sponsorships – an estimate of the maximum height scaled, even if an expedition fails? These are calculated through similar structures called regression trees (as opposed to classification trees which we have been using here), one instance of which is shown through the last row of Table 1.

In the long history of predictive modelling, these trees are relatively new developments. But their dominance is now almost total. Whatever the obstacles, a kind of unwavering determination somehow endures. With mountaineers, it's their will to climb Everest. With modellers, these trees. ■

Note

An interactive dashboard featuring all peaks above 6000 metres is available at moinak.shinyapps.io/MountainExpeditions. Find codes for this Everest analysis at https://github.com/moinakbhaduri/ModellingEverest

Reference

1. Friedman, J. H. (2002) Stochastic gradient boosting. *Computational Statistics & Data Analysis*, **38**(4), 367–378.