Pick a Tree – Any Tree

Alex Chin, Gary Gordon, Kellie MacPhee, and Charles Vincent

Abstract. A subtree is picked at random from the collection of all subtrees of a complete graph. What is the probability the subtree spans? We find the surprising answer this question and a few closely related questions.

1. INTRODUCTION. Trees are important in a variety of applications within mathematics and the sciences. Indeed, MathSciNet lists nearly 18,000 articles with the word "tree" in the title. To a mathematician, a *tree* is a connected graph with no cycles.[1]

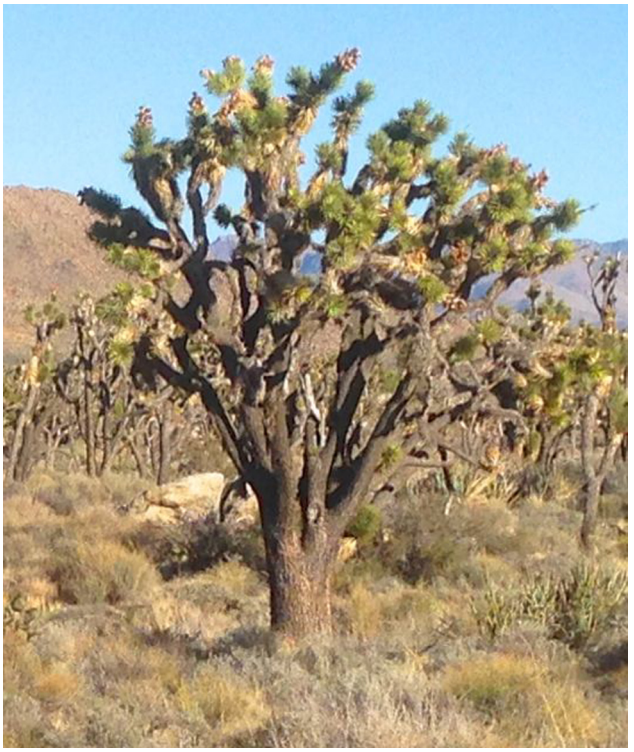

Figure 1. This Joshua tree lives in the Mohave desert.

Our motivation here is the following, easy-to-digest question.

Question 1. Suppose we pick a tree, at random, among all the subtrees (spanning and nonspanning) in a complete graph. What are the chances that it is a spanning tree?

http://dx.doi.org/10.4169/amer.math.monthly.122.5.424
MSC: Primary 05C05

[1] Banyan trees and some other tropical trees can have cycles. The authors know of no botanical examples where trees are disconnected, although banana trees may be joined underground.

For a quick example, consider the complete graph K_4 (see Figure 2). Of the 38 subtrees of various sizes, there are a total of 16 spanning trees. If we select a subtree at random, where each subtree has an equal chance of being selected (*uniform probability*), then the probability our subtree spans is $\frac{16}{38} \approx 42\%$.

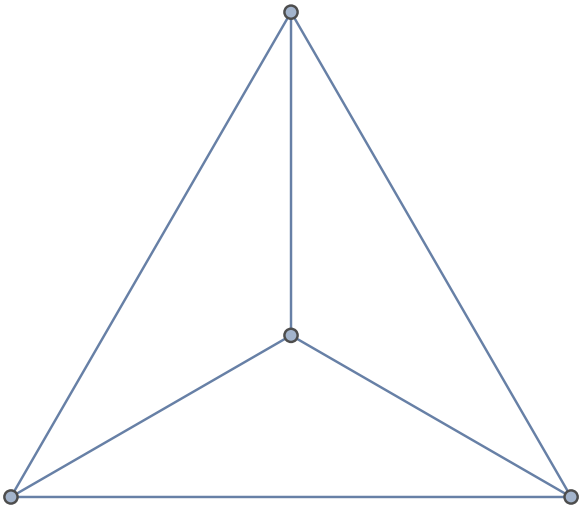

Figure 2. The complete graph K_4 has four subtrees with one vertex (the four vertices), six subtrees with two vertices, 12 subtrees with three vertices and 16 spanning trees using all four vertices.

We seek the asymptotic value of the limit: If p_n is the probability we have selected a spanning tree of the n-vertex complete graph K_n, we are after $\lim_{n\to\infty} p_n$. We answer this question completely in Theorem 4 at the end of Section 3. Like most mysteries, you can ruin the surprise by skipping ahead. But there is a surprise waiting for you there.

Why should you care about this? Well, here are a few reasons.

- Trees are fundamental structures in graph theory. Spanning trees in graphs give minimal ways to connect networks. Moreover, finding minimum weight spanning trees in a network is the prototypical example of when the *greedy algorithm* produces an optimal solution to a problem (see Section 2).
- Counting problems are fun. Most (discrete) probability problems are counting problems in disguise, and our problem can be phrased in a purely combinatorial way.
- Many (most?) mathematical questions, even well-motivated, interesting ones, have a narrow appeal to specialists working in the area. Often, even the best questions have answers that are not very interesting. But in this case, we are lucky: The motivation behind our question is easy to understand, and the answer is interesting and surprising.
- Solutions to typical problems often rely on results that are treated as a black box: "We use Theorem X to prove Theorem Y." This is how math gets done, and it would be impossible to prove many (any?) of the deep theorems in mathematics from scratch. But here, again, we're lucky. The solution is easy to follow, requiring nothing more than first year calculus.
- Finally, even when the question and answer are easy to understand, the proof may not shed any light on why the answer is what it is. For instance, one of the basic

proof techniques of combinatorics and graph theory is induction, which is very useful but requires you to know the answer before you start. Here, the answer pops out of the analysis.

It's always a good idea to get some data. The probabilities for some values of $n \leq 100$ are given in Table 1.

Table 1. Probabilities of selecting a spanning tree in K_n. Do you recognize the limit?

n	p_n
10	0.617473
20	0.657876
30	0.669904
40	0.675689
50	0.679090
60	0.681329
70	0.682915
80	0.684097
90	0.685012
100	0.685741

In determining the limiting value of the probability, we'll encounter the number e three times.[2] The key facts we'll need are a very important, famous formula credited to Arthur Cayley, one of the giants of 19th century English mathematics, and two standard facts familiar to all calculus students.

This paper is organized as follows. Section 2 introduces Cayley's formula, and solves a warm-up problem.

> Fix a vertex v in the complete graph K_n, and randomly choose a spanning tree T. What is the probability v is a leaf in T?

Section 3 outlines a proof of the main result (Theorem 4), although we sweep several details under the rug. We conclude with some related questions and a few exercises in Section 4.

2. CAYLEY'S FORMULA AND A WARM-UP PROBLEM. Suppose your boss gives you the following job: Find the cheapest way to connect 100 cities using telephone wires.[3] You know how much it costs to connect each pair of cities. Immediately, you realize that your network needs to be connected, but cycles are unnecessary.

This is the standard *minimum weight spanning tree* problem for a graph, and it has an interesting history. Two slightly different greedy algorithms solve this problem completely, *Kruskal's algorithm* and *Prim's algorithm*. As is often the case in mathematics, however, the discoveries were initially made by others. In this case, Czech mathematicians Otakar Borůvka (1926) and Vojtěch Jarník (1930) showed that greedy algorithms always find the cheapest spanning trees in any graph. J. B. Kruskal (1956) rediscovered Borůvka's algorithm and R. C. Prim (1957) rediscovered Jarník's. Kruskal and Prim were both working for Bell Labs, so the problem was not solely of academic interest. See [**4**] for an account of the history of this problem.

Greedy algorithms are fast because they avoid backtracking. The greedy algorithm does what you might guess; you choose the cheapest "legal" edge you can (where

[2] That's nothing compared to the number of times we'll encounter the *letter* e.

[3] Your boss thinks it's 1953, evidently.

"legal" means avoiding cycles), and repeat until you have a spanning tree. But you would like to impress your boss (usually a good idea), so you decide to first tell her how many potential solutions there are.

In 1889, Arthur Cayley published a beautiful, simple formula counting the number of spanning trees in the complete graph K_n.

Theorem 1 (Cayley's Theorem). *The number of spanning trees of K_n is n^{n-2}.*

Cayley was not the first person to discover this formula; J. J. Sylvester published an equivalent formula in 1857, and C. W. Borchardt published the formula in 1860. Cayley acknowledged Borchardt's work in his 1889 paper, but Cayley extended the formula and phrased it in modern graph-theoretic terms.

Cayley's formula tells us how hopeless it is to examine all possible spanning trees for a large complete graph. For instance, if we wish to connect 100 cities with wires using the architecture of a tree, it would be impossible to look at each of the 100^{98} possibilities. For reference, the number of atoms in the observable universe is estimated to be approximately 10^{80}. At minimum, your boss should give you extra time to complete this project.[4]

There are several beautiful proofs of Cayley's formula. Aigner and Ziegler highlight three different proofs in their entertaining book [**1**], and Moon gives several distinct proofs in [**5**]. One standard proof, attributed to Prüfer,[5] establishes a bijection between sequences of length $n-2$ and spanning trees of K_n.

To get the sequence (called the *Prüfer code*) for a labeled tree, do the following.

1. Find the leaf with the greatest label (a vertex is a *leaf* if it touches just one edge.)
2. Write down the label of the vertex adjacent to your leaf, then remove that leaf (*pruning the tree*).
3. Repeat until only two vertices remain.

The process is completely reversible, and every possible sequence of length $n-2$ can occur. This requires proof, but it also gives us our bijection.

Since there are n^{n-2} such sequences, there are also n^{n-2} labeled trees on n vertices.

Exercise 1. Find a tree whose Prüfer code is your phone number.

Now notice that the number of times any label appears in the code is one less than the degree of the corresponding vertex. For instance, in Figure 3, the vertex labeled 5 has degree 4, and the label 5 appears three times in the code. This observation leads to our first connection with a calculus problem.

Question 2. Fix a vertex of K_n. What is the probability that the vertex is a leaf of a randomly chosen spanning tree of K_n?

Solution. To find the number of spanning trees that have vertex 1 (say) as a leaf, we simply count the number of Prüfer codes that do not use the label 1. We still need to create a sequence of length $n-2$, but now there are only $n-1$ symbols available. So we get $(n-1)^{n-2}$ spanning trees where vertex 1 is a leaf.

[4] Of course, the point of using the greedy algorithm is that you don't need to look at all 100^{98} spanning trees to find the cheapest one: The greedy algorithm produces the cheapest very rapidly. But you don't need to tell your boss that.

[5] You might think that Prüfer is the origin of the word *proof* in mathematics. You would be wrong.

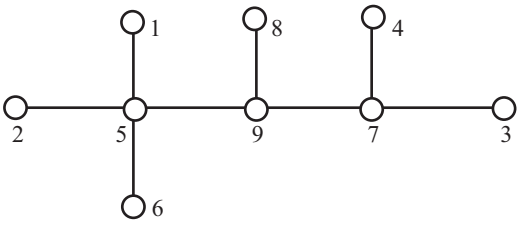

Figure 3. A spanning tree of K_9 with Prüfer code 9577955

Since the total number of spanning trees is n^{n-2}, we find the probability that vertex 1 is a leaf is

$$\left(\frac{n-1}{n}\right)^{n-2}.$$

Then L'Hôpital's rule (applied to the log of this expression) gives a limiting probability of $1/e = 0.3678\ldots$. ■

Most calculus texts give some of the historical background on L'Hôpital and his relationship with the Bernoullis. Although the rule was discovered by Johann Bernoulli, it is called L'Hôpital's rule because of a "curious business arrangement" the two men agreed upon. In fact, L'Hôpital acknowledged Bernoulli's discovery of the rule, but Bernoulli believed he did not receive sufficient credit.

As a final application of this procedure, here's another homework problem.

Exercise 2. What is the limiting probability that neither vertices 1 nor 2 are leaves in a randomly chosen spanning tree of K_n?

3. THE PROBABILITY OUR SUBTREE SPANS. We return to our main question: What is the limiting probability that a randomly chosen subtree of K_n is a spanning tree? To compute our probability, we will need to count the number of subtrees a_k in K_n with k vertices, for $1 \le k \le n$. But this is easy.

- Choose k vertices in $\binom{n}{k}$ ways.
- Create a tree using these vertices in k^{k-2} ways (Cayley's formula).

Putting these two steps together gives us the number of subtrees with k vertices:

$$a_k = \binom{n}{k} k^{k-2}.$$

Now we need to estimate the probability p_n of picking a spanning tree:

$$p_n = \frac{\#\text{ spanning trees}}{\text{total }\#\text{ of trees}} = \frac{a_n}{a_n + a_{n-1} + \cdots + a_1}.$$

We know $a_n = n^{n-2}$. To estimate this fraction, we will concentrate on the denominator $a_n + a_{n-1} + \cdots + a_1$. The key to estimating this sum is to look at the ratio of consecutive terms. Assuming $1 \le k < n$, a little algebra gives us

$$\frac{a_{k+1}}{a_k} = \frac{\binom{n}{k+1}(k+1)^{k-1}}{\binom{n}{k}k^{k-2}} = (n-k)\left(\frac{k+1}{k}\right)^{k-2}.$$

If k is reasonably large, then we can estimate $\left(\frac{k+1}{k}\right)^{k-2}$, again using L'Hôpital's rule. (This particular limit is important because it gives the effective interest rate for continuous compounding.) Then $\lim_{k\to\infty}\left(\frac{k+1}{k}\right)^{k-2} = e \approx 2.71828$. If you're keeping score, this is our second encounter with e.

Although these terms approach e, they converge rather slowly. Here are the first 10 values:

k	$\left(\frac{k+1}{k}\right)^{k-2}$
1	0.5
2	1.
3	1.33333
4	1.5625
5	1.728
6	1.85262
7	1.94966
8	2.02729
9	2.09075
10	2.14359.

We may (somewhat safely) conclude the following.

Proposition 2. For $k < n$, with k sufficiently large, we have $a_{k+1} \approx e(n-k)a_k$.

Applying Proposition 2 repeatedly gives us a recursive way to compare the size of a_{n-k} with a_n. In particular, we see

$$a_n \approx e \cdot 1 \cdot a_{n-1} \approx e^2 \cdot 1 \cdot 2 \cdot a_{n-2} \approx \cdots \approx e^k k! a_{n-k}.$$

We summarize this relationship with a proposition.

Proposition 3. Let a_m be the number of subtrees of K_n that have exactly m vertices, for $1 \le m \le n$. Then $a_{n-k} \approx \frac{a_n}{k!e^k}$.

We're almost done. We can now use Proposition 3 to estimate $\sum_{k=1}^{n} a_k$:

$$(a_n + a_{n-1} + \cdots + a_1) \approx a_n\left(1 + (1/e) + \frac{(1/e)^2}{2!} + \cdots + \frac{(1/e)^k}{k!} + \cdots\right).$$

This sum should remind you of a very familiar Taylor series. The series $e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!}$ is probably the most important power series you encounter in calculus. If we evaluate[6] at $x = (1/e)$, we get

$$a_n + a_{n-1} + \cdots + a_1 \approx a_n\left(e^{(1/e)}\right).$$

Then the limiting probability is

$$p_n \approx \frac{a_n}{a_n e^{(1/e)}} = (1/e)^{(1/e)} = 0.692201\ldots.$$

Hence, we have the answer to our main question.

[6]This is our last encounter with e, and note that e appears twice here: in the Taylor series and the evaluation.

Theorem 4. *Choose a subtree at random from the family of all subtrees of the complete graph K_n. Then the probability that the subtree is a spanning tree approaches $(1/e)^{(1/e)} = 0.692201\ldots$ as $n \to \infty$.*

Truth in advertising note: Our proof isn't a complete proof. We've omitted the details about how accurate the approximations are, but they do work out. See [**2**] for the missing pieces.

4. MORE QUESTIONS, ANSWERS, AND SURPRISES. The choice to use a uniform probability is somewhat arbitrary. What if we change the rules for picking subtrees? Instead of each subtree having an equal chance of being chosen (so the probability of choosing a subtree with only one edge is the same as the probability of choosing a spanning tree, for example), a natural choice is to *weight* the probability by the number of edges the tree contains. In this scenario, what happens to the probability of selecting a spanning tree?

Question 3. Suppose we pick a tree, at random, among all the subtrees (spanning and nonspanning) in a complete graph, where the probability of choosing a subtree is proportional to the number of edges in the subtree. What are the chances it spans?

Before giving the answer, we give a famous example to illustrate the difference between these two probability measures.

All schools report "average class size." There are two popular ways to do this.

1. First choose a student at random, then ask her to randomly choose one of her classes.
2. List all classes taught at the school, then randomly choose one of them.

Colleges and universities typically measure "average class size" using the latter technique. Why? It turns out that this always gives a lower average! This was first noticed by Feld and Grofman in 1977 in [**3**] in an article in an education journal. This curious fact also explains why your Facebook friends seem to have more friends than you do, and why almost everyone believes they are one of the weakest people at the gym.

In our application, think of the classes as the subtrees and edges as the students. Although we are not concerned with the average size of a subtree here, it's easy to use Theorem 4 to show the "average" subtree of K_n is a spanning tree.

Returning to Question 3, we call our unweighted probabilities p_n and our weighted versions q_n. As expected, weighting subtrees by their size increases the chances of selecting a spanning tree, i.e., $p_n < q_n$. Table 2 gives data for these values when $n \leq 100$.

The rather surprising fact is that in the limit, weighting doesn't matter: the limiting probabilities are the same!

Theorem 5. *Let p_n be the uniform, unweighted probability of choosing a spanning tree of K_n, and let q_n be the weighted probability of choosing a spanning tree (with weight proportional to the number of edges of the subtree). Then*

$$\lim_{n\to\infty} p_n = \lim_{n\to\infty} q_n = e^{-e^{-1}} = 0.692201\ldots.$$

Table 2. Probabilities of selecting a spanning tree using uniform and weighted probabilities.

n	p_n	q_n
10	0.617473	0.652736
20	0.657876	0.672725
30	0.669904	0.679294
40	0.675689	0.682552
50	0.679090	0.684497
60	0.681329	0.685789
70	0.682915	0.686711
80	0.684097	0.687401
90	0.685012	0.687936
100	0.685741	0.688365

Proof sketch. As with p_n, we can get a simple expression for q_n:

$$q_n = \frac{na_n}{na_n + (n-1)a_{n-1} + \cdots + 2a_2 + a_1}.$$

Let $b_k = ka_k$. Then, arguing as we did for p_n, we get

$$b_{n-k} \approx \left(\frac{n-k}{n}\right)\frac{b_n}{k!e^k}.$$

Then $\displaystyle\sum_{k=0}^{\infty} b_{n-k} \approx b_n e^{1/e}\left(1 - \frac{1}{en}\right)$, which gives $q_n \to e^{-e^{-1}}$.

Here are two more things for you to try at home.

Exercise 3. Prove the fact that average class size is greater when reported by students than it is when reported by the registrar.

Exercise 4. As a final challenge, try computing the limiting probability for the complete bipartite graph $K_{n,n}$. Here are the data for values of $n \le 1000$.

n	p_n
50	0.475495
100	0.477349
150	0.477953
200	0.478253
250	0.478432
300	0.478551
350	0.478636
400	0.478699
450	0.478749
500	0.478788
1000	0.478965

Can you guess the limit? It might be useful to note that $\left((1/e)^{(1/e)}\right)^2 \approx 0.479142$.

[Hint: The number of spanning trees in the complete bipartite graph $K_{r,s}$ is $r^{s-1}s^{r-1}$. Let t_k be the number of subtrees of $K_{n,n}$ *avoiding* exactly k vertices. Then show the ratio $t_k/t_0 \approx \frac{2^k}{e^k k!}$ when n is large.]

ACKNOWLEDGMENT. This research was supported by NSF grant DMS-1063070.

REFERENCES

1. M. Aigner, G. Ziegler, *Proofs from the Book*. Fourth edition. Springer-Verlag, Berlin, 2010.
2. A. Chin, G. Gordon, K. MacPhee, C. Vincent, Subtrees of complete graphs, submitted, ArXiV version http://arxiv.org/abs/1308.4613.
3. S. L. Feld, B. Grofman, Variation in class size, the class size paradox, and some consequences for students, *Research in Higher Education*, **6** (1977) 215–222.
4. R. L. Graham, P. Hell, On the history of the minimum spanning tree problem, *Ann. Hist. Comput.* **7** (1985) 43–57.
5. J. W. Moon, Various proofs of Cayley's formula for counting trees, *A Seminar on Graph Theory*, Holt, Rinehart and Winston, New York, 1967. pp. 70–78.

ALEX CHIN is a Ph.D. student in statistics at Stanford University. He is interested in combining mathematical, statistical, and computational approaches to solving problems. Before moving to California, he studied mathematics, economics, and linguistics at North Carolina State University. He enjoys hiking, cooking, and playing the piano.
Department of Statistics, Stanford University, 390 Serra Mall, Stanford, CA 94305
ajchin@stanford.edu

GARY GORDON received his B.A. from the University of Florida in 1977 and his Ph.D. from the University of North Carolina in 1983. He ran Lafayette College's REU program for a decade and has thoroughly enjoyed doing research with enthusiastic undergraduates. He is currently the Problem Editor for *Math Horizons*. He loves baseball, enjoys climbing things, and, like Alex, occasionally plays the piano.
Mathematics Department, Lafayette College, Easton, PA 18042
gordong@lafayette.edu

KELLIE MACPHEE is currently a Ph.D. student in the Mathematics Department at the University of Washington. She recently graduated from Dartmouth College with a major in mathematics and a minor in Japanese, and her mathematical research interests are in combinatorial optimization. In her free time, she enjoys various athletic activities like running and playing water polo.
Mathematics Department, University of Washington, Seattle, Wa 98105
kmacphee@math.washington.edu

CHARLES VINCENT is currently a Ph.D. student in mathematics at the University of Iowa. He recently completed his undergraduate degree in mathematics at Lafayette College in Easton, PA. When not engaged in schoolwork, he likes to travel the world and play the saxophone. He is also that kid who doodles irrelevant nonsense in a notebook during meetings.
Mathematics Department, University of Iowa, Iowa City, IA 52242-1419
charles-vincent@uiowa.edu