CS 4510

Kolmogorov Complexity

Name: Abrahim Ladha

1.1 Book

Please read through section 6.4, pages 261-269 of the Sipser book[1] before continuing. The book refers to $K(x)$ as minimal description, but this can be confused with something from machine learning, so we denote it as Kolmogorov complexity

1.2 Proofs

Theorem 1.1 *There does not exist a computable function for* $K(x)$

Proof: Suppose $K(x)$ is computable. Lets contruct the following TM M: Lexographically for each $x \in \Sigma^*$, check if $K(x) > |\langle M \rangle|$, and return the first x. This program outputs the shortest x such that $K(x) > |\langle M \rangle|$, but M prints x, and is exactly size $|\langle M \rangle|$, a contradiction. ■

I think this proof is really beautiful because it is so simple. Here $|\langle M \rangle|$ is hardcoded into the machine, but if you want a stronger idea, you may apply the recursion theorem from 6.1 of Sipser, which states that any machine has access to its own encoding.

Theorem 1.2 $R = \{x \in \Sigma^* \mid K(x) \geq |x|\}$ *is undecidable*

Proof: Suppose not. Then there exists a TM M which decides R. We construct the following machine M'. It takes as input an integer n, and for each $x \in \{0,1\}^n$ lexographically, returns the first x such that $M(x)$ says $x \in R$. Then $K(x) \geq n$, but we can represent x in $\log n$ bits, so $K(x) < \log n + c$, which for large enough n is not true. ■

We have shown that the imcompressible strings are undecidable, but it is also true that the compressible strings are undecidable, by a similar proof.

1.3 Proof by Method of Incompressibility

Kolmogorov complexity can be a very useful, if niche, part of your theorem proving toolkit. Here are some applications.

Theorem 1.3 $L = \{0^n 1^n \mid n \in \mathbb{N}\}$ *is not regular.*

Proof: Suppose L is regular, then it has some DFA D to decide it. Suppose on input 0^i we end on state q. Then on input a series of $1s$, we will be in the accept state if and only if we input exactly i 1s. We then have a description of i, as the DFA plus the state q, so $K(i) \leq |D| + c$. Choose some i such that $K(i) \geq \log i + c$, and then we have a contradiction for sufficiently large i. ■

[1] If you need a copy send me an email

Theorem 1.4 (Euclid's Theorem) *There are infinitely many primes*

You are probably familar with the classical euclidean proof, but here is one of a different kind.
Proof: Suppose not. Let $p_1, ..., p_k$ be finitely many primes. Let x be an incompressible string, and let n be the integer representation of x. Then there exists $e_1, .., e_k$ such that $n = p_1^{e_1} ... p_k^{e_k}$. Each $e_i < \log n$ so $|e_i| < \log \log n$, but then $K(x) < k \log \log n + c$, a contradiction since we assumed x to be incompressible. ■

Theorem 1.5 *Any TM to decide* $\{xx^R \mid x \in \Sigma^*\}$*, the palindromes, takes* $\Omega(n^2)$

We won't prove the theorem, here, but in another worksheet. It is an interesting result since the first naive way to decide the palindromes is $O(n^2)$, so this says we can't do much better.

1.4 Problems

Turn in 4 of 5

1. Can you give two complex strings x_1, x_2 such that $x_1 + x_2$ is simple? Here addition is bitwise, like integers.
2. Give an upper bound to the ratio of strings of length n have the property that $K(x) \leq |x| - 3$?
3. Give a tight upper bound on $K(x^n)$ for x any string as a function of x and n. (Hint: Sipser covers the case $K(xx)$)
4. Give a computable monotonic and decreasing function such that $\lim_{t \to \infty} f(t, x) = K(x)$.
5. Let C, C' be a lossless compression algorithm and the corresponding decompression algorithm, and define *x.zip* as the output of running C on x. Give a lower bound on the length of *x.zip.zip....zip* in terms of n, x and some constant independent of both, where n is the number of "zips".

Further Reading

[1] Ming Li and Paul M. B. Vitányi. "A New Approach to Formal Language Theory by Kolmogorov Complexity". In: *CoRR* cs.CC/0110040 (2001). URL: https://arxiv.org/abs/cs/0110040.

[0] Michael Sipser. *Introduction to the Theory of Computation*. Cengage learning, 2012.

[0] Ming Li, Paul Vitányi, et al. *An introduction to Kolmogorov complexity and its applications*. Vol. 3. Springer, 2008.

[0] Lance Fortnow. "Kolmogorov complexity". In: *Aspects of Complexity, Minicourses in Algorithmics, Complexity, and Computational Algebra, NZMRI Mathematics Summer Meeting, Kaikoura, New Zealand*. Vol. 317. 2000.

[0] Luca Bortolussi. *A short overview of Kolmogorov complexity and its application for proving lower bounds of algorithms*.