Classifying ultraconserved elements with hidden Markov model of evolutionary profiles

Name + Affiliation Removed

Abstract

Ultraconserved elements (UCEs) are DNA segments that are observed to be near 100% identical across all vertebrates including humans. Despite millions of years of evolution, natural selection has selected for these sequences to remain unchanged, yet the functions of UCEs remain unknown. It is theorized that the edges of UCEs contain palindromic DNA sequences that may contribute to DNA structure regulation. The first step to validate this hypothesis is to formulate a model to classify where the UCE edges are. In this study, I constructed a hidden Markov model trained on evolutionary conservation profiles generated by bioinformatics software PhyloP and Phastcon to identify locations of UCEs and their edges on the human genome. My results demonstrate the usefulness of combining HMM with conservation profiles, and can be applied across entire human genome to find novel UCEs. These findings form the basis to testing the palindromic hypothesis, and further allow geneticists to carry out additional downstream sequence analysis on UCEs to decipher their potential function(s).

1 Background

While this paper is ideally geared towards readers immersed in the field of bioinformatics, critical biological details necessary for understanding the paper are briefly highlighted. Readers are strongly encouraged to explore the relevant cited references if they feel overwhelmed by the biology mentioned.

1.1 What are UCEs and why should we care

Almost all life forms on Earth contain DNA encoding genetic instructions necessary for carrying out survival functions. DNA is composed of 4 types of nucleotides: adenine (A), thymine (T), cytosine (C), and guanine (G) [1]. Evolution theory tells us that all modern species share a common ancestry over ~3.9 billion years ago [1]. If a segment of DNA is conserved between multiple species (ex. shared between humans, dogs, cats and rats), this segment of DNA most likely carries out an important biological function [1,2]. This is indeed true for many DNA sequences that code for proteins, where small mutations in these regions can lead to diseases like breast cancer and Parkinson disorder. Therefore, throughout the course of evolution, functional sequences are preserved with minimal changes. Conversely, if a DNA segment is not important for survival, we would expect to see that segment mutating and diverging over evolution time, and not conserved across species (ex. polymorphisms in humans) [2].

Ultraconserved elements (UCEs) are DNA segments observed to be highly conserved from fish to primates [3]. They are first categorized in 2004 by Haussler et al., where a genomic comparison between human, mouse and rat genomes reveal certain segments of perfect conservation (i.e. 100% identity). No other DNA elements have such high degree of conservation [3]. This remarkable conservation implies they carry out certain functions in our body that make them very important (hence conserved). Mathematically speaking, if we assume UCEs do not carry out any function, then the chance of observing UCEs is estimated to be less than 10^{-22} [3]. However, after 7 years of intense study, we still have little conclusion as to what functions these regions' functions have, or what they do in our body. Their biological roles remain a mystery.

To date, 481 UCEs have been identified in the human genome [3]. These are identified by simply searching for segments of human genome that show 100% identity with corresponding segments of genomes in rodents. Such method fails to address the following issues: 1) it does not incorporate level of conservation in other species like fish. UCEs exhibit high level of consideration across all known animals, and this method only looks at rodents and human. 2) It can only roughly pinpoint where UCEs are, but is unable to define the

specific boundaries that mark the start and end of UCEs.

1.2 Project goal

This study marks the starting point of testing the hypothesis that UCEs play a role in gene regulation (i.e. regulating when and how much of a protein to be produced in the body at a given time). Our DNA is organized as a double-helix with 4 strands, 2 from the mom and 2 from the dad. I hypothesize that the 4 DNA strands at the UCE edges (edge defined as the boundary between a UCE and a non-UCE) possess palindromic sequences such that they form cross-overs where one DNA strand from one copy interacting with a DNA strand from another copy, and vice versa [see reference 4 for more details on cross-overs]. Cross-overs have been studied intensively in meiosis, mitosis and many other forms of structural variants where they play a role in gene regulation [4], but not in UCEs. If my theory is correct, I should see palindromic nucleotide sequences located at the edges of the UCEs.

Prior method for detecting UCEs is incapable of pinpointing precisely where UCEs begin and end; one cannot look for palindromic sequences in the edges until one can first reliably call where these edges are. In this research, I employ a hidden Markov model (HMM) trained on evolutionary conservation data to characterize the precise start and end windows for UCEs. Once the UCEs edges are identified, this will facilitate genetic researchers to explore the sequence properties of the edges and see if palindromic sequences are indeed present, leading to great potential impacts for our understanding of the role(s) UCEs have in our body.

Throughout the rest of paper, I define an UCE edge to be a window of 15 nucleotides, flanking the body of UCE and separating UCEs from non-UCEs. This number is arbitrarily chosen based on prior biological knowledge.

2 Materials and Method

2.1 Materials

A list of 481 known UCEs is obtained from Bejerano et al. [3]. The setup of HMM is facilitated by the General Hidden Markov Model library (GHMM, *http://ghmm.org/).* The evolutionary profiles used to train and test the HMM are downloaded from UCSC genome database [5]. These profiles are generated at the primate level, mammal level (which also includes primate species), and vertebrate level (which includes both primate and mammal species) via bioinformatics software PhyloP [5] and Phastcon [5].

2.2 Why HMM

The goal of using machine learning in this project is to accurately classify where UCEs edges are. There are at least over 481 UCEs in the human genome, and to manually classify all 481 edges is an exhausting task, while a machine learning approach will greatly streamline the process.

Hidden Markov models (HMMs) have been widely deployed in bioinformatics from protein sequence alignment, protein family annotation, to gene-finding [6]. A HMM consists of M states that are used to annotate an input sequence. (ex. a string of DNA) [6]. The assignments of those states to each nucleotide (i.e. the classification) depend on a set of emission probabilities and transition probabilities. The prediction output is a state path with the highest overall probability, the so-called optimal state path, or Viterbi path [6].

HMM in bioinformatics is especially powerful when there is an underlying biological structure to capture. For example, in gene-finding application, HMM allows researchers to capture the various components that make up a gene: promoters, introns, exons, splice junctions. Each position along the input sequence is dependent on the annotation assigned to the previous position(s). The nature of HMM makes it ideal for one to setup models to capture the dependency and structure of the underlying data, which would be difficult if working with other types of machine learning algorithms. For example, if one is to use multinomial classifier or random forest to classify nucleotides as UCE, edge, or non-UCE, it will be arduous to incorporate the dependencies between nucleotides with such algorithm, whereas the HMM makes this a straightforward process.

2.2 Training and testing datasets

My training dataset consists of 100 known UCEs [3] plus a window of 200 nucleotides flanking both sides of each UCE to capture the neighboring non-conserved nucleotides. Within this set, each UCE and its flanking sequences are manually annotated by a trained geneticist, who assigned a label of "non-UCE", "edge" or "UCE" to each nucleotide. A nucleotide that is assigned "non-UCE" represents a nucleotide that is not part of the UCE. The nucleotide that is assigned "edge" represents the nucleotide that is part of the edge flanking one of the two sides of UCEs. The nucleotide that is assigned "UCE" represents the nucleotide that makes up the core body of the UCE. The size of test set is the remaining 381 known UCEs not part of the training set. In the initial phase of the project, no manually annotation was given to the test set, but later on, for performance evaluation purposes, the geneticist manually examined the predictions made by the HMM on the test set and assessed the correctness.

Since UCEs are defined by high level of conservation across species, I trained and tested my model on evolutionary profiles. Each UCE has 6 associated evolutionary conservation profiles. They are generated by programs PhastCons and PhyloP. Each program outputted 3 conservation profiles, corresponding to conservation for 8 primates, 31 placental mammals, and 44 vertebrates (see Materials). PhastCons works by fitting a phylo-HMM to the genomes by maximum likelihood and considers neighboring bases, while PhyloP score is a separate measurement of conservation that examines each base independently without taking neighbors into account. I considered different levels of conservation (i.e. primates, mammals, vertebrates) because from a biological perspective, one would expect human DNA to exhibit higher conservation with rest of primates compared to entire vertebrates. Incorporating varying conservation levels allow me to gain higher resolution in looking for conserved sequences between human genomes and rest of vertebrates.

The input to the HMM can be visualized as multiple matrices, where each matrix represents a UCE. The columns correspond to each nucleotide along the DNA sequence, and the rows correspond to 6 different levels of evolutionary conservation. For the training dataset, there is an additional row that specifies what annotation (i.e. non-UCE, edge, UCE) each nucleotide is assigned to in order to train the model's parameters.

2.3 Model setup

I construct my HMM to reflect the underlying biological structure of UCEs. Figure 1 below depicts a graphical representation of the way my model is setup.

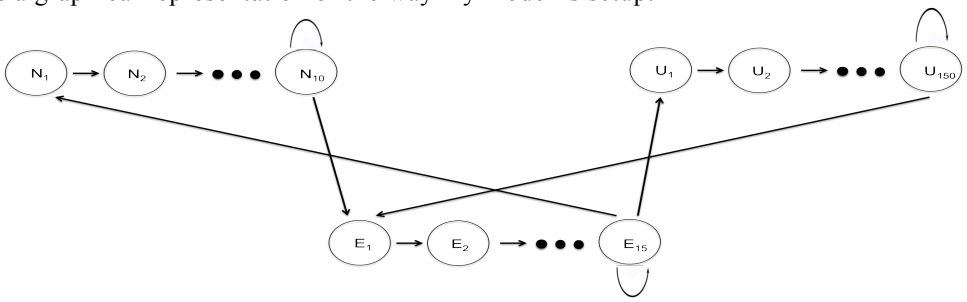

Figure 1. Graphical representation of the HMM. States labeled as N represent non-UCE states, E represent edge states, and U represent UCE states. The subscript represents the number of states for each type in total (ex. 10 states in total for modeling non-UCE regions). The arrows represent the allowed transition. State emissions, start and end states are not shown in the figure.

The non-UCE states model the stretches of nucleotides that correspond to portions of DNA not part of UCE. I restrict the non-UCE states transition probability between each other to be 1 for 10 consecutive transitions so that a non-UCE region has to be at least 10 nucleotides long. This is to reflect the biological property that non-UCE regions occur in continuous stretches of nucleotides, not short broken fragments. The edge states model the regions that correspond to UCE edges. Here I force an UCE edge to be at least 15 nucleotides or longer by having 15 edge states connecting to each other with transition probability 1, again to reflect prior biological knowledge of how UCE edges look like based on prior manual inspection. For UCE states, I force them to model at least 150 nucleotides. Previous studies [3-4] have used 200 nucleotides as the minimum length threshold, but here I am relaxing the threshold to accommodate for the lengths taken up by the edges. Some states like E_{15} and U_{150} are allowed self-transitions to capture edges and UCEs longer than 15 and 150 nucleotides respectively. My model forbids a transition from non-UCE to UCE to ensure edge states must be read before progressing from a non-UCE region to an UCE region. The start and end states of the model are not labeled, but essentially any state can be transitioned from the start (with a small arbitrary probability), and any state

can be transitioned to the end (also with a small arbitrary probability). The start and end states represent the beginning and end of the input.

The output of my HMM across each UCE sequence is a string of state annotations, with a label assigned to each nucleotide (among the three possible labels: non-UCE, edge, UCE). The nucleotides assigned as "edge" correspond to the edges for the UCEs. The assignment is done by looking at the Viterbi path [7], the state path through the entire input sequence with the highest overall probability. The path depends on the probabilities assigned to the transition and emission parameters, which are trained with an expectation maximization (EM) algorithm discussed below.

2.4 Training the parameters with Baum-Welch algorithm

The Baum-Welch algorithm is an expectation-maximization algorithm used to train the emission and transition parameters within a HMM [7]. It defines an iterative procedure in which the emission and transition probabilities in iteration n + 1 are set to the number of times each transition and emission is expected to be used when analyzing the training sequences with the set of emission and transition probabilities derived in the previous iteration n. The following section highlights the key points of the algorithm and how it is applied in this project.

Let $T_{i,j}^{n}$ denote the transition probability for going from state i to state j in iteration n. An example would be transition from non-UCE state to edge state. I fixed transition probabilities between the states of the same type (ex. non-UCE to non-UCE, or edge to edge) to be 1, only the transition probabilities between states of different types are trained with Baum-Welch algorithm. The transition probability models the likelihood to see a certain evolutionary score based on the state you are at. For example, transitioning into a UCE state when evolution score is low (i.e. low conservation) is unlikely, as reflected by the annotations provided in the training set.

Let $E_i^n(y)$ denote the emission probability for emitting letter y in state i in iteration n. In my project, y is the output prediction assigned to each nucleotide (ex. a non-UCE state assigns a non-UCE label). For my project, emission probability is trivial because each state only emits 1 possible output.

Let P(X) be the probability of getting sequence X, and x_k be the k^{th} letter in input sequence X. We also define Xk as the sequence of letters from the beginning of sequence X up to sequence position k, (x_1, ...x_k). X_k is defined as the sequence of letters from sequence position k + 1 to the end of the sequence, (x_{k+1}, ...x_L), where L is the length of sequence X.

For a given set of training sequences, S, the expectation maximization update for transition probability $T_{i,j}^{n+1}$ can then be written as $T_{i,j}^{n+1} = \frac{\sum_{X \in S} t_{i,j}^{n}(x) / P(X)}{\sum_{j'} \sum_{X \in S} t_{i,j}^{n}(X) / P(X)}$, where

$t_{i,j}^{n}(x) := \sum_{k=1}^{L} f^{n}(X_k, i) T_{i,j}^{n} E_j^{n}(X_{k+1}) b^{n}(X^{k+1}, j)$. The superfix n on the right hand side indicates the quantities are based on the transition probabilities $T_{i,j}^{n}$ and emission probabilities $E_i^n(x_{k+1})$ of iteration n. f(Xk, i): = P(x_1, ...x_k, s(x_k) = i) is the so-called forward probability of the sequence up to and including sequence position k, requiring that sequence letter x_k is read by state i. It is equal to the sum of probabilities of all state paths that finish in state i at sequence position k. The probability of sequence X, P(X), is therefore equal to f(XL, End). b(Xk, i): = P(x_{k+1}, ...x_L|s(x_k) = i) is the so-called backward probability of the sequence from sequence position k + 1 to the end, given that the letter at sequence position k, x_k, is read by state i. It is equal to the sum of probabilities of all state paths that start in state i at sequence position k. The forward and backward probabilities fn(Xk, i) and bn(Xk, i) can be calculated using the forward and backward algorithms [7].

The expectation maximization update for emission probability $E_i^{n+1}(y)$ is carried out with the same logic as updating transition probability, see [7] for more details.

3 Results

3.1 Performance on known UCEs

I evaluated my HMM performance on the 381 already-known UCEs left out during training. Prior studies on these UCEs only have the main body annotated, but the locations of the edges remain uncertain, so I collaborated with a trained geneticist to compile a training set with edges manually annotated. The performance is evaluated by sensitivity, defined as the proportion of nucleotides that have their states correctly assigned. Table 1 below summarizes the performance:

	Sensitivity performance on 381 UCEs at nucleotide level
Non-UCE	84.5%
Edge	72.1%
UCE	95.8%
Overall	84.1%

Table 1. Performance evaluation on the known UCEs. Sensitivity is calculated by dividing the number of nucleotides correctly predicted over the total number of nucleotides belonging to that category. Performance is noticeably worse for predicting the edges, as expected due to the varying nature inherent in the edges.

Overall, my HMM yields a welcoming performance above 84%, but this performance is not evenly distributed among the three states. The HMM is able to find the main UCE body very decently, with over 95% of the nucleotides correctly assigned. Likewise for the non-UCEs, with over 84% correct predictions. The lowest performance comes from predicting the edges. This is due to the varying nature of the edges. Generally the edge is the region where there is a change in conservation profile from low to high (if going from non-UCE to UCE), and vice versa (if going from UCE to non-UCE). This degree of change, or the slope, is not the same among edges. Some edges have a sharp slope, which in this report, I will address them as strong edges. Others have a gentler slope, which I address as medium edges. There are edges, which I address as weak edges, where the slope is so gentle it is hard even for a trained biologist to determine precisely where the edges are. It is this third class of edges that is resulting in the most miss-classification (see Table 2 below).

	Sensitivity performance on 381 UCEs for specific edge types
Strong edge	89.4%
Medium edge	73.4%
Weak edge	59.7%
Redundant edges	52.4%

Table 2. Performance evaluation based on class-specific edges. The strong edges that exhibit a strong cutoff from non-UCEs in terms of evolutionary profile are most easily identified. Weak edges have evolutionary profiles that are hard to differentiate from non-UCEs, even by eye. Redundant edges are the most poorly captured among all edge types.

It is apparent that my misclassification arises mostly from the edges that have a gentle conservation slope, and hence, can be hard to identify even by eye. There are also some UCEs that have redundant edges, where two or more edges occur in proximity to each other instead of simply just one on each side. Although my model does allow the capture of such edges, performance reveals very poor classification on the edges not directly next to UCEs. This is probably due to lack of such examples in the training set. Increasing the number of such cases in the training set should improve this performance.

3.2 Removal of PhyloP tracks boost performance

I next see if I can improve the performance by removing some of the evolutionary conservation profiles. Figure 2 summarizes the effects of removing PhyloP tracks on performance.

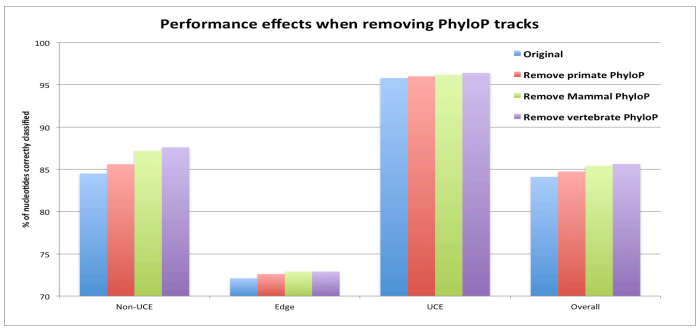

Figure 2. Performance is increased when PhyloP tracks are removed indpendently of other tracks. The removal is done by simply not feeding it to the model. The opposite is observed when removing Phastcon tracks (not shown).

It is apparent that PhyloP information does not help in classifying UCEs. The tracks only add noise, which in retrospect is expected because this information is generated by only looking at single-column conservation, without taking into account of the neighboring nucleotides. I repeated the same experiment with Phastcon scores, the performance is found to decrease when removing the Phastcon information (results not shown). This illustrates that in this situation, Phastcon information is much more useful at predicting UCEs because the scores are generated by taking into account of neighboring nucleotides.

4 Conclusions

The function(s) of UCEs remain a mystery; no labs have yet conclusively determined the biological role(s) of these DNA sequences. There is a theory that UCE edges contain palindromic sequences in order for DNA to form alternative structure instead of the standard double-helix, resulting in a different DNA folding that ultimately impact gene regulation.

4.1 Significance

Identifying where these edges are serves as the basis for testing such theory. In this paper, I applied HMM to classify locations of UCEs and their edges. Trained on evolutionary conservation profiles, the algorithm yields decent performance and is able to recapture over 72% of the known nucleotides identified as UCE edges. This result is significant to those working in the genetics field. My study differs from previous attempts at finding UCEs by incorporating evolutionary conservation profiles across all vertebrate species. My HMM is also able to locate edge regions that previous methods fail to address. One bioinformatics downstream analysis is to analyze the nucleotide composition of those edges, and see if they indeed contain palindromic sequences that are over-represented compared to other parts of the genome. Also, it will be interesting to explore other nucleotide compositions within these edges, such as dinucleotides AA/TT, which is often associated to rigidity in DNA structure. Additionally, researchers can also apply my model across the whole human genome and identify potentially novel UCEs.

4.2 Improvements

There remains several improvements to boost the HMM. As previously discussed, the results show that most of the misclassification is due to diverse edge properties, where some UCEs exhibit a strong differentiation between UCE and its edge, while others are less obvious. Further improvements can be made to address this issue by expanding two new group of edge states, so there are three sets of edge states in total. This essentially will model the UCE edges as three separate classes: one that is a weak edge, a medium edge, and a strong edge. Changing the arbitrary lengths defined in the model (ex. edge length from 15 to 13) may also improve the performance. Furthermore, in this study, I only used evolutionary profiles as training and testing, but there are many other features to feed into the model as well. For example, UCE edges are previously known to be enriched in adenine and thymine, incorporating such property into my classifier can boost the performance. Additionally, other features such as the information of whether a nucleotide is located within a protein-coding region, or whether the nucleotide is known to be epigenetically modified, can also be included to further differentiate UCEs and their edges from non-UCEs. Lastly, some UCEs exhibit redundant edges where an edge is followed by another edge. My HMM currently does poorly in predicting the latter edges because my training data lacks such example. Increasing the size of my training set, combined with the novel features mentioned above, should better capture this type of edge.

5 References

[1] Cracraft, J.; Donoghue, M.J., eds (1995) Assembling the tree of life. Oxford University Press. Pp. 592.
[2] Sawyer SA, Parsch J, et al.(2007) Prevalence of positive selection among nearly neutral amino acid replacements in Drosophila. Proc. Natl. Acad. Sci. USA. **104**(16):6504-10.
[3] Gill Bejerano, Michael Pheasant et al. (2004) Ultraconserved elements in the human genome. Science Vol.304 pp.1321-1325.
[4] Masatoshi Nei. (1987) Molecular Evolutionary Genetics. ISBN 0-231-06320-2.
[5] Paulin A. Fujita et al. (2011) UCSC Genome Browser database: update 2011. Nucleic Acids Research. Vol. 39 pp.876-882.
[6] Schuster-Bockler B.,et al. (2007) An introduction to hidden Markov models. Curr Protoc Bioinformatics. Appendix 3A.
[7] Istvan Miklos, Irmtraud Meyer. (2005) A linear memory algorithm for Baum-Welch training. BMC Bioinformatics. 6:231.