

# Dendrochronology: Algorithmically Cross Dating Tree Rings

William Heaton

23 Sept 2012

## **Motivation**

As trees grow their yearly growth leaves a pattern of differential colorations known as tree rings. One ring is produced for each year of growth of the tree. Features of these rings can provide useful information about the conditions in which a tree was growing including historical climate data. A further feature of these tree rings is that within regions of similar climate, they can be compared to one another. This comparison allows researchers to overlay and match tree ring series from different trees. For example, you could compare a 100 year old tree which spanned years 1850-1950 with a more recent tree which spanned 1900-2012. From this comparison you could determine the exact years that the first tree was alive without having previous knowledge of those dates. And you can continue this process dating tree rings back up to thousands of years if the wood is preserved. The purpose of this project is to provide automated, statistically meaningful, and powerful tree ring cross dating software to the Dendrochronology community. To do this we employ algorithms originally made for pattern matching of DNA sequence and DNA map information in computational biology.

## **Pairwise Alignment**

Thanks to the field of computational biology especially Waterman and others, there exists a considerable literature on pairwise map alignments.[7][2][6][3] They are similar to algorithms for pairwise sequence alignment originally by Needleman and Wunsch[4] and

extended by Smith and Waterman.[5][1] In the example below the horizontal lines represents the two Tree Series cores and the tick marks represent tree ring divisions. Here you can see the distance between each tick represents the tree rings. Further, the dotted lines between ticks on the other strand represent the alignment of those rings. The ordered set of pairs of probes aligned in the optimal alignment are the pairwise alignment between two tree ring series For notation we will denote  $v_j^i$  as the  $j^{th}$  ring division on tree ring series  $i$ .

Figure 1: Pairwise Map Alignment.

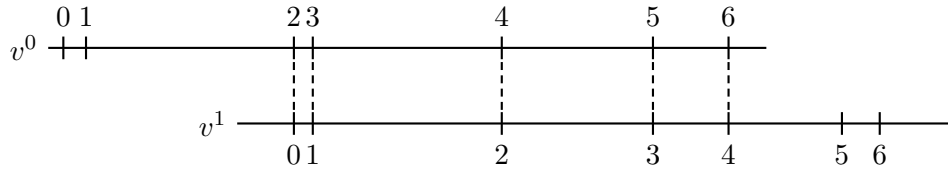


Figure 2: Giving the ordered set of pairs for this alignment as  $\{\{v_2^0, v_0^1\}, \{v_3^0, v_1^1\}, \{v_4^0, v_2^1\}, \{v_5^0, v_3^1\}, \{v_6^0, v_4^1\}\}$

## Normalization

Different tree's grow at different rates whether that is a constant multiple on the size of each tree ring and/or a constant addition to each year's growth. Another bias that must be dealt with is that most tree's follow an exponential decay in growth from year to year. We wish to compare tree rings from different trees at any time in their life cycle. Previous methods employed regression fits of various types (exponential, spline) to subtract off the trend of ring sizes over the course of a tree's lifespan. Here I propose a more general way of normalizing tree ring data. Trends of any type serve to explain the low frequency variation of tree ring widths. I propose to remove the low frequency variation directly via a high pass filter on the discrete Fourier transform of the tree ring data and then normalize the resulting inverse fourier transform data to a mean of 0 and standard deviation of 1. This also provides us with a meaningful and straight forward way of identifying outliers. At a later date I will provide some graphs and graphics of tree rings before and after their

normalization to demonstrate the improvement in comparability.

## Distributions

Need to work on figuring this out more. What are the tree ring sizing distributions? And what are they post-normalization? What are the error distributions and do they correlate in some way with the tree ring sizes (more or less error compared to the ring sizes).

## Error Model

I will get to this later. There is tree ring sizing error as well as missing rings in both the master series and the raw tree ring measurements. The master series has much less error due to improvements made via averaging over many tree ring series data. Basically I am currently using a sum of squares scoring function and which works for most distributions but isn't be the best for any nor is it particularly meaningful.

## Missed Rings

Rings are missed (not seen, not measured) with some probability. We shall refer to missing rings from now on as **false negatives**. One question I have regarding false negatives is “Does the probability of a tree ring missing depend on the size of the tree ring?” Master series can also have missed rings but at a much lower probability.

It is worth noting a few things about false negatives in conjunction with positional error. When a tree ring series contains a false negative it may be ambiguous which year is missing  $\pm 1$  year. The reason for this is that I am about to fall asleep so I should do this later but I have in mind a nice figure that explains it visually. It is also important to mention that when false negatives occur very close to the beginning or end of the tree's lifespan that missed ring may be overlooked by the algorithm. The reason for this is that under tolerated sizing error, it may be a better explanation of the data to say that no false negative occurred but that the last (or first) few tree rings were not closely matched. You could even imagine a situation in which, by happenstance those intervals do match remarkably well. Once again this is not of large concern as it does maintain the overall alignment and we

will have statistically meaningful metrics of whether the overall alignment matches or does not match two Tree series (or a series and a master series).

Figure 3: Pairwise Alignment Ambiguities

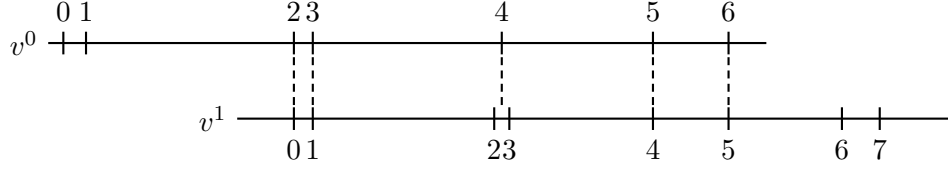


Figure 4: Here it is unclear whether the tree ring separation  $v_4^0$  should be aligned with  $v_2^1$  or  $v_3^1$ . An exact tie is unlikely but sizing error can cause pairwise alignment will pick the incorrect ring to gap (in that one ring is matched to a ring from which year it did not arise). This is not a problem because it is the most likely ring to have been gapped under the error model and still maintains the overall alignment.

---

**Algorithm 1** Alignment Algorithm

---

```

procedure ALIGN( $X, Y$ )
   $\forall i M_{i,0} \leftarrow 0$ 
   $\forall j M_{0,j} \leftarrow 0$ 
  for  $i \leftarrow 0, n$  do
    for  $j \leftarrow 0, m$  do  $\triangleright X_i$  and  $Y_j$  are the right sides of the interval
       $maxScore \leftarrow -\infty$ 
      for  $k \leftarrow i - 1, \max(0, i - \delta)$  do  $\triangleright$  Looking back for left end of interval
        for  $l \leftarrow j - 1, \max(0, j - \delta)$  do
           $g \leftarrow i - k + j - l - 2$   $\triangleright$  gapped rings
           $tmpScore \leftarrow M_{i,j} + S(X_{k,i}, Y_{l,j}, g)$ 
          if  $tmpScore > maxScore$  then
             $B_{i,j} \leftarrow [k, l]$ 
             $maxScore \leftarrow tmpScore$ 
          end if
        end for
      end for
       $M_{i,j} \leftarrow maxScore$ 
    end for
  end for
end procedure

```

---

---

**Algorithm 2** Viterbi backtrace of dynamic programming matrix

---

```
procedure BACKTRACE(B)
   $maxEnd \leftarrow -\infty$ 
  for  $i \leftarrow 0, n$  do                                 $\triangleright$  Find max score on the end edges on which to backtrace
    if  $B_{i,m-1} > maxEnd$  then
       $k, l \leftarrow i, m - 1$ 
       $maxEnd \leftarrow B_{i,m-1}$ 
    end if
  end for
  for  $j \leftarrow 0, m$  do
    if  $B_{n-1,j} > maxEnd$  then
       $k, l \leftarrow n - 1, j$ 
       $maxEnd \leftarrow B_{n-1,j}$ 
    end if
  end for

  while  $k \geq 0 \& l \geq 0$  do                                 $\triangleright$  Backtrace
    add  $k, l$  to alignment
     $k, l \leftarrow B_{k,l}$ 
  end while
end procedure
```

---

## References

- [1] Richard Durbin, Sean Eddy, Anders Krogh, and Graeme Mitchison, *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. cambridge univ, 1998.
- [2] X. Huang and Michael S. Waterman, *Dynamic programming algorithms for restriction map comparison*, Computer Applications in the Biosciences **8** (1992), no. 5, 511–520.
- [3] Niranjana Nagarajan, Timothy D. Read, and Mihai Pop, *Scaffolding and validation of bacterial genome assemblies using optical restriction maps*, Bioinformatics **24** (2008), no. 10, 1229–1235.
- [4] S. B. Needleman and C. D. Wunsch, *A general method applicable to the search for similarities in the amino acid sequence of two proteins.*, Journal of molecular biology **48** (1970), no. 3, 443–453.

- [5] T. F. Smith and M. S. Waterman, *Identification of common molecular subsequences.*, Journal of molecular biology **147** (1981), no. 1, 195–197.
- [6] Anton Valouev, Lei Li, Yu-Chi Liu, David C. Schwartz, Yi Yang, Yu Zhang, and Michael S. Waterman, *Alignment of optical maps*, Journal of Computational Biology **13** (2006), no. 2, 442–462.
- [7] Michael S. Waterman, Temple F. Smith, and Harold L. Katcher, *Algorithms for restriction map comparisons*, Nucleic Acids Research **12** (1984), 237–242.