2009-03-04

Characterizing the Statistical Properties and Global Distribution of Dansgaard-Oeschger Events

Andrea Michelle Thomas
Brigham Young University - Provo

CHARACTERIZING THE STATISTICAL PROPERTIES AND GLOBAL DISTRIBUTION OF DANSGAARD-OESCHGER EVENTS

by

Andrea Lundrigan Thomas

A thesis submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Master of Science

Department of Statistics

Brigham Young University

April 2009

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a thesis submitted by

Andrea Lundrigan Thomas

This thesis has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

Date	William F. Christensen, Chair
Date	Summer Rupper
Date	Scott D. Grimshaw

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the thesis of Andrea Lundrigan Thomas in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

Date

William F. Christensen
Chair, Graduate Committee

Accepted for the Department

Scott D. Grimshaw
Graduate Coordinator

Accepted for the College

Thomas W. Sederberg
Associate Dean, College of Physical and Mathematical Sciences

ABSTRACT

CHARACTERIZING THE STATISTICAL PROPERTIES AND GLOBAL DISTRIBUTION OF DANSGAARD-OESCHGER EVENTS

Andrea Lundrigan Thomas

Department of Statistics

Master of Science

Ice core records from Greenland have shown times of rapid warming during the most recent glacial period, called Dansgaard-Oeschger (D-O) events. D-O events are important to our understanding of both past climate systems and modern climate volatility. In this paper, we present new approaches for statistically evaluating the existence of cyclicity in D-O events and the possible lagged correlation between the Greenland and Antarctica temperature records. Specifically, we consider permutation testing and bootstrapping methodologies for assessing the cyclicity of D-O events and the correlation between the Greenland and Antarctica records. We find that there is not enough evidence to conclude that D-O events are cyclical; however, the Antarctica record leads the Greenland record by 545 years with a statistically significant correlation of 0.455.

ACKNOWLEDGEMENTS

With my heart full of gratitude, I thank my husband and family for their undying support. Because they believe in me, I was able to find the courage to press forward. I thank Forrest for rejoicing with me in my success, being a shoulder to cry on when everything seemed impossible, and encouraging me to pursue my highest aspirations. I thank Mom for reminding me that I can do hard things, proof reading countless papers at a moment's notice, and the tireless reassurance and emotional support. I thank Dad for the confidence in my ability, debugging code for me at 2:00 am numerous times, and the sound advice about attitude. Thanks to Rachel, Levi, and Philip for the genuine interest and invaluable friendship.

I am grateful to Dr. Scott for encouraging me to learn more about the integrated master's program, Dr. Grimshaw for caring about my success and general well-being, Dr. Rupper for the expert advice on my research and genuine enthusiasm, and especially Dr. Christensen for the patience, support, and friendship as my advisor. I thank the entire graduate faculty for their contribution to my education and achievement. I also thank the Rocky Mountain NASA Space Grant Consortium and the BYU Statistics Department for funding my research.

Most importantly, I thank my Father in Heaven for His merciful aid. With His strength and goodness, I am able to complete the impossible. In every success I achieve, I gladly acknowledge Him.

CONTENTS

TABLES

FIGURES

1. INTRODUCTION

Ice cores from various locations of the world are extremely useful in evaluating global climate change. The primary ice cores used in this research come from Greenland and Antarctica. From these ice core records, chemists determine the ratio of certain isotopes, such as oxygen (δ^{18}O) and deuterium (δD), found at certain depths. The depths at which the isotopes are extracted indicate at what time in the earth's history the isotopes were preserved in the ice. These isotopes are useful because they are indicative of the temperature at the time they were preserved. The ice core records from Greenland have shown times of rapid warming during glacial periods, called Dansgaard-Oeschger (D-O) events (Dansgaard et al. 1993).

The scientific community is divided about certain characterizations of D-O events. Some have argued that a regular process paces D-O events and that D-O events are therefore cyclical in nature (Alley and Clark 1999; Schulz 2002; Rahmstorf 2003). It can be inferred from this theory that D-O events have a predictive element, which may shed light on the current climate change the earth is experiencing. Contrary to this theory, others believe that D-O events are random in their timing. Statistical analyses have not been performed with the newest and highest quality ice core records in relation to this controversy. As a result, conclusions about the nature of D-O events and their potential connection to current climate change cannot be affirmed.

The other disagreement relates to the geographic extent of D-O events. Some are convinced that occurrences similar to D-O events are discernible in the Antarctica ice core records (Dansgaard et al. 1993; Blunier et al. 1998; EPICA Community Members 2006; Jouzel et al. 2007; Steffensen et al. 2008). A connection of this sort between Antarctica and Greenland fosters the hypothesis that the climate in the

Northern and Southern Hemispheres is linked by some mechanism, oceanic or atmospheric (Dansgaard et al. 1993; Blunier et al. 1998; EPICA Community Members 2006; Steffensen et al. 2008). Others speculate that the events seen in the Antarctica ice core records have no temporal connection to D-O events (Roe and Steig 2004; Ackert Jr. et al. 2008). Under this theory, it is suggested that separate regional forces govern the climate of the Northern and Southern Hemispheres and that occasionally, climate events occur that are large enough to be recorded on a global scale (Roe and Steig 2004). Despite this difference in theory behind the presence of D-O events on a global scale, no one in the scientific community has statistically assessed these arguments using the most recent and highest quality ice core records.

The purpose of this thesis is to evaluate the stochastic nature of D-O events and whether or not D-O events occur on a global scale. Specifically, we investigate if D-O events are cyclical or if they occur randomly during the most recent ice age. Understanding D-O events throughout the earth's history will shed more light on the current climate changes we are experiencing. In addition, we explore whether or not D-O events are evident in the Antarctica ice core record or found just in the Greenland ice core record. The presence of D-O events in both records may signify a relationship between climates in the Northern and Southern Hemispheres. The converse points to independent climate in the North and South that is determined mostly by regional influences.

2. LITERATURE REVIEW

The first objective of this thesis is to assess the nature of D-O events; specifically, we intend to determine if D-O events are cyclical in nature or if they are random variations of the climate. Establishing the nature of D-O events is significant because it will provide more insight as to how the climate functions.

Alley and Clark (1999) discuss the speculated stochastic nature of D-O events. They suggest that the variability of the climate, as demonstrated through D-O events, is linked to a mechanism driving changes in the circulation of the Atlantic Ocean. Changes in this circulation have a cyclical nature and thus Alley and Clark (1999) assert that D-O events are also cyclical. They conclude that their assertions will require further study and analysis.

Schulz (2002) evaluates the nature of D-O events and determines that D-O events between 46 to 13 kyr before present (BP, where present is 1950) were driven by a stochastic process that occurred roughly every 1,470 years. The Greenland record beyond 50 kyr BP contains uncertainty in dating such that Schulz (2002) cannot make conclusions regarding the stochastic nature of D-O events found beyond 50 kyr BP. Schulz (2002) uses a trapezoidal wave model created by D-O events 5 through 7 to determine the regularity of D-O events. Between 46 to 13 kyr BP, 7 out of 8 D-O events matched his trapezoidal model. The probability of this happening by chance is about 1%. Based on this result, Schulz (2002) determines that D-O events from 46 to 13 kyr BP were not random in nature but occurred approximately every 1,470 years.

Rahmstorf (2003) proposes the concept of a cyclical clock that paces D-O events as opposed to the idea that D-O events are random. D-O events are present approximately every 1,470 years during the last glacial period. Occasionally, the time between

two D-O events is 3,000 or 4,500 years. Rahmstorf (2003) argues that this pacing is too regular to be considered random. He suggests that a cyclical process takes place every 1,470 years that triggers D-O events. If a D-O event is not triggered when it should have been according to this clock, it is because the mechanism that triggers D-O events failed and not the underlying clock. Although the cause of this pacing is unknown, because of the regularity of D-O events Rahmstorf (2003) hypothesizes that it is due to external sources, such as variations in the sun or the orbit of the earth, rather than internal causes, such as the circulation in the Atlantic Ocean.

Alternatively, Huybers and Curry (2006) suggest that stochastic resonance partially explains climate variability. Specifically, they propose that climate variability is determined by a stochastic component in addition to a deterministic component. Huybers and Curry (2006) state that understanding the deterministic element of climate variability will lead to an improved ability to predict climate volatility.

The second objective of this thesis is to determine if the rapid climate change of D-O events can be found in the Antarctica ice core records or if these events are exclusive to Greenland records. Establishing the presence or absence of D-O events in ice core records from Antarctica is important because it signifies either a relationship between the climate in the Northern and Southern Hemispheres or regionally independent climate systems.

Dansgaard et al. (1993) identify and define D-O events as rapid climate changes during glacial periods found in the Greenland ice core records. They determine that the presence of D-O events is proof that the North Atlantic region climate can transform quickly, leading to climate variability. They compare these newly defined D-O events to records from Antarctica and find that similar events occurred in Antarctica during the last glacial period. The events found in the Antarctica ice core records are less frequent and less extreme than D-O events. Dansgaard et al. (1993) attribute this difference to a mechanism that connects the Northern and Southern Hemispheres: the

rapid ocean and atmospheric circulation changes found in the North Atlantic. Essentially, they argue that this mechanism influences the Southern Hemisphere but with less intensity than it does the Northern Hemisphere.

Blunier et al. (1998) compare the Greenland ice core records to the Antarctica ice core records and find that Greenland lagged Antarctica by approximately 1 to 2.5 kyr over the period 37 to 23 kyr BP. They contend that because Antarctica leads Greenland, the hypothesis that the Northern Hemisphere influences warming in the Southern Hemisphere can be rejected. Blunier et al. (1998) also assert that the length of the lag implies that the Northern and Southern Hemispheres cannot be connected through the atmosphere; instead, they propose that the Northern and Southern Hemispheres are linked through the Atlantic Ocean.

A new ice core, EDML, was drilled from Antarctica, which rendered better comparisons between ice core records from Greenland and Antarctica because of the high quality of this new core. EDML is easier to compare directly to the records from Greenland and aids in the pursuit of locating D-O events in Antarctica. The EPICA Community Members (2006) compare Greenland ice core records to EDML and find that events of a similar magnitude as D-O events are apparent in EDML, but slightly before D-O events present themselves in the Greenland ice core records. They attribute this to a bipolar seesaw, which is to say: when the Southern Hemisphere cools the Northern Hemisphere typically warms. Specifically, they assert that temperatures in the Northern Hemisphere lag temperatures in the Southern Hemisphere and thus the Southern Hemisphere influences the climate of the Northern Hemisphere through a mechanism in the Atlantic Ocean.

Steig (2006) discusses the findings of the EPICA Community Members (2006) in his paper. He indicates the importance of the new ice core from Antarctica and the benefit of being able to compare this high-quality core to Greenland's well-dated records. Steig (2006) also explains briefly the process of how the Southern Hemisphere

could influence the climate in the Northern Hemisphere through a mechanism in the Atlantic Ocean, called the deep meridional overturning circulation. Even though he is enthusiastic about EDML and the potential it has, Steig (2006) is wary of believing the assertions made by the EPICA Community Members without substantial proof that the climate in the Northern and Southern Hemispheres are related.

A new technology called continuous flow analysis has been developed that allows a higher resolution of the chemical records from ice cores to be obtained. This technology was applied to the NGRIP ice core record from Greenland. Steffensen et al. (2008) consider the transition between the last glacial period and the current interglacial period using this higher resolution version of NGRIP. From this examination, they determine that the warming transition occurring at 14.7 kyr BP happened within three years. Steffensen et al. (2008) hypothesize that this extremely rapid warming indicates that the climate is sensitive to the Northern Hemisphere atmospheric circulation and the Atlantic meridional overturning circulation and that the climate between the Northern and Southern Hemispheres is linked.

Supporting the Hemispheric climate connection but on a millennial time scale, Jouzel et al. (2007) assess the relationship between the Greenland and Antarctica ice core records and find that these records are connected. They recognize characteristics in the Antarctica records that are similar to the Greenland records during the glacial periods, although these events are smoother and less abrupt than D-O events. Jouzel et al. (2007) determine that both sets of records show large scale millennial variability; however, the Antarctica records do not necessarily demonstrate all of the climate variability contained in the Greenland records.

Bond et al. (1993) argue that D-O events are found in locations of the North Atlantic region other than Greenland. They suggest that D-O events are large enough that other locations in the Northern Hemisphere would also experience these events. Bond et al. (1993) compare ice core records from Greenland with ocean sediment

records from the North Atlantic region and discover the records from the North Atlantic Ocean contain events correlated to D-O events. Although this thesis does not directly consider the presence of D-O events in other regions of the Northern Hemisphere, naturally, if these events are observed in Antarctica they would also be observed in other areas of the Northern Hemisphere besides Greenland.

Roe and Steig (2004) primarily use the ice core records BYRD and GISP2 from Antarctica and Greenland, respectively, to evaluate the possible presence of D-O events in Antarctica. Their results yield no substantial relationship between the climate in the Northern and Southern Hemispheres. This article describes theories about the relationship between the climate in the Northern and Southern Hemispheres, such as a lag/lead relationship, and then systematically provides evidence disproving these theories based on the comparison of BYRD and GISP2. Roe and Steig (2004) conclude that the ice core records from the Northern and Southern Hemispheres occasionally demonstrate events of similar magnitudes around the same relative time period, but they attribute this to climate events that were so globally prevalent they would have inevitably been recorded in both records. They find that the climate in the Northern and Southern Hemispheres is independent of one another and is determined primarily by regional influences.

Ackert Jr. et al. (2008) explore the possibility of the Younger Dryas as a global event. The Younger Dryas is a cooling period that took place during the last deglaciation identified in North Atlantic paleoclimate records. Although the Southern Hemisphere experienced a cooling period preceding the Younger Dryas, Acker Jr. et al. (2008) evaluate if this Southern Hemisphere cooling was the result of a North Atlantic signal or was caused by regional forces. After studying glacier retreats from various lakes in the Southern Hemisphere, they find that the cooling experienced in the South was not due to a North Atlantic signal. Rather, they conclude that the climate between the Northern and Southern Hemispheres is governed primarily by

regional influences.

Although the articles just discussed provide good insight into the issues of interest, some authors failed to provide statistically based conclusions while others used ice cores that are of a lesser quality than the ones now available. With this research, we use ice cores that have been widely accepted as being of a higher quality than other cores, and importantly, we draw all conclusions based on formal statistical analysis. The statistical methods utilized in this research involve permutation testing and bootstrapping methodologies.

Pardo-Igúzquiza and Rodríguez-Tovar (2000) suggest using a permutation test to assess the significance of a power spectrum. Specifically, they suggest that a random permutation of observations within a series should destroy any cyclicity contained in the series. Through simulation, the cyclicity found in the new series can then be compared to the cyclicity observed in the original series to evaluate if what was observed could have happened by chance. Similarly, Swanepoel (1986) proposes applying bootstrapping methods to the power spectrum to obtain confidence intervals. The ideas expressed in this article can be extended to nonparametric hypothesis testing. Both Pardo-Igúzquiza and Rodríguez-Tovar (2000) and Swanepoel (1986) indicate that their methods are especially appealing because they are nonparametric and therefore, distributional properties are not assumed. We likewise perform a permutation test and use bootstrapping approaches when assessing the power spectrum. With our research, we do not perform a traditional bootstrap. Instead of resampling each observation with replacement, we resample groups of observations dependent over time with and without replacement. Also, we perform 10 times the number of simulations than originally proposed by Pardo-Igúzquiza and Rodríguez-Tovar (2000).

Block bootstrap methodologies are useful when data contain dependencies over time (see Hall 1985; Carlstein 1986; Künsch 1989; Politis and Romano 1994; Christensen and Sain 2002). Specifically, Christensen and Sain (2002) demonstrate that

using statistical inference tools that do not account for dependencies when dependencies exist in the data can render the results invalid. They develop a methodology for incorporating dependencies using a nested block bootstrap for multivariate modeling. Likewise, it is essential that we take into account the dependencies inherent in the Greenland and Antarctica ice core records. For our research, the blocks created depend on the data; specifically, each D-O event creates a block. Because this method is not entirely a block bootstrap approach, we refer to it simply as bootstrapping. Since we account for the dependencies within the data, the inferences presented in this thesis are valid.

3. PAPER TO BE SUBMITTED TO *NATURE*

3.1 Abstract

Ice core records from Greenland have shown times of rapid warming during the most recent glacial period, called Dansgaard-Oeschger (D-O) events. D-O events are important to our understanding of both past climate systems and modern climate volatility. In this paper, we present new approaches for statistically evaluating the existence of cyclicity in D-O events and the possible lagged correlation between the Greenland and Antarctica temperature records. Specifically, we consider permutation testing and bootstrapping methodologies for assessing the cyclicity of D-O events and the correlation between the Greenland and Antarctica records. We find that there is not enough evidence to conclude that D-O events are cyclical; however, the Antarctica record leads the Greenland record by 545 years with a statistically significant correlation of 0.455.

3.2 Introduction

Greenland ice core records show times of rapid warming during glacial periods, called Dansgaard-Oeschger (D-O) events (Dansgaard et al. 1993). Some argue that a regular process paces D-O events and D-O events are therefore cyclical in nature (Alley and Clark 1999; Schulz 2002; Rahmstorf 2003). In this manuscript, we assess this hypothesis to better understand the appearance of D-O events. It can be inferred that if D-O events are cyclical in nature, then they contain a deterministic component (Huybers and Curry 2006). The possibility of cyclicity in D-O events may be further explored to better understand climate variability in general. If D-O events instead appear to be randomly distributed throughout time, then the timing of such extreme

warming events is not easily characterized and is not likely to be associated with well-documented astronomical or geophysical cycles.

Another characteristic of D-O events that is uncertain relates to the manifestation of D-O events on a global scale. Some suggest that occurrences similar to D-O events are discernible in the Antarctica ice core records (Dansgaard et al. 1993; Blunier et al. 1998; EPICA Community Members 2006; Jouzel et al. 2007; Steffensen et al. 2008). A connection of this sort between Antarctica and Greenland fosters the hypothesis that the climate in the Northern and Southern Hemispheres is linked by some mechanism. Others speculate that the events seen in the Antarctica ice core records have no association with D-O events (Roe and Steig 2004; Ackert Jr. et al. 2008). Under this theory, it is suggested that regional forces govern the climate in the Northern and Southern Hemispheres and that occasionally, climate events occur that are large enough to be recorded on a global scale (Roe and Steig 2004).

The purpose of this paper is to evaluate the stochastic nature of D-O events and whether or not D-O events occur on a global scale. Specifically, we investigate if D-O events exhibit a statistically significant cyclical component or if they are randomly distributed events. In addition, we explore if D-O events are evident in the Antarctica ice core records or found only in Greenland.

3.3 Methods

An initial examination of GISP2 and EDML revealed a quadratic trend in the data. To eliminate this problem, both records were detrended and the analysis was performed on the residuals of the detrended data. Because of the nature of ice cores, GISP2 and EDML contain unevenly spaced observations. In addition to this, GISP2 is of a higher resolution than EDML, presenting difficulties when comparing GISP2 to EDML. In order to remedy these issues, GISP2 and EDML were linearly interpolated to be of the same resolution with evenly spaced observations. For our analysis, two

series were considered. The first series, referred to as the "short series," ranges from 47,245 to 20,000 year BP and contains D-O events 2 through 13. Because the first D-O event occurs during the beginning of the transition between the glacial and interglacial periods, there is not a general consensus as to if this really is a D-O event. To be thorough, we considered the time frame that included this event as well, which ranges from 47,245 to 12,500 years BP. This time frame will be called the "long series." The short series actually introduces a bias towards causing D-O events to appear cyclical because we consider this tight window in which D-O events are consistently appearing, excluding time periods where no D-O events are observed. Because the long series includes D-O event 1, and thus a stretch of time when no D-O events are observed, this series does not contain the same bias towards making D-O events appear cyclical.

The occurrence of D-O events can be viewed from two perspectives, either non-overlapping or overlapping in nature. These two perspectives are referred to as the non-overlapping D-O events model (NOM) and the overlapping D-O events model (OM). The NOM implicitly assumes that a D-O event is characterized by a sharp increase in temperature followed by a sustained and gradual return to baseline temperature. Because of this rigid definition of a D-O event, a second D-O event cannot occur until the first has run its full course. The OM assumes that a D-O event is characterized by the initial sharp increase in temperature, with the gradual return to baseline temperature being a natural result of the initial change in temperature. Thus, the OM allows for a D-O event to occur before a previous D-O event has run its full course. We consider both characterizations of the D-O event.

Under the OM two perspectives can be taken: (1) D-O events 8 and 12 are not the result of overlapping D-O events but are simply longer than other D-O events, or (2) D-O events 8 and 12 may be considered to be periods in which more than one warming event occurs so closely in time that the temperature does not return

to baseline between events. The second perspective implies that D-O events 8 and 12 should not be included in the data generation for the analysis. We explored the potential for cyclicity under both positions and found the conclusions to be the same. For the purposes of this manuscript, we discuss the findings of the first perspective when referencing the OM.

3.3.1 Cyclicity Detection

In order to assess the cyclicity of D-O events, permutation testing and bootstrapping methodologies were applied. Our end goal with these methods is to create a distribution of temperature records that represent the range of possible records under the hypothesis of no cyclicity. Specifically, the D-O events within each series were randomly shuffled (sampled without replacement) or resampled (sampled with replacement) to destroy any cyclicity among the observed sequence of D-O events. Although the p-values associated with shuffling D-O events were typically lower, the conclusions reached when shuffling or resampling D-O events were the same. For simplicity, the results presented in this manuscript are derived from resampling D-O events only. D-O events were identified objectively using a piece-wise linear model algorithm. The observations between D-O events, referred to as the "filler," were randomized such that groups of sequential observations also remained together in the new series. It was important to allow the filler to be randomized such that continuous segments were retained in order to preserve some of the dependencies inherent to the time series. These resulting series will be referred to as pseudo-series, which act as possible realizations of temperature records under the assumption of the hypothesis of no cyclicity. These pseudo-series were linearly interpolated to contain 1,000 observations. As a result, comparisons can be made between the observed, linearly interpolated series and the pseudo-series without affecting statistical significance.

By the use of periodograms, the maximum periodogram peak of the pseudo-

series was obtained and compared to the periodogram peaks associated with the observed series. The periodogram peak of the observed series is the test statistic. The filler of the observed series was also randomized using an approach that resembles the block bootstrap (Hall 1985; Carlstein 1986; Swanepoel 1986; Künsch 1989; Politis and Romano 1994; Christensen and Sain 2002), but ensuring that we have removed any residual low-frequency cyclicity that may be associated with the filler. For the evaluation of each model (NOM and OM), periodogram peaks associated with very low-frequency cycles (i.e., cycles with periods exceeding 2,500 or 3,000 years) were treated as unrelated to D-O event-like phenomena. We refer to this upper bound for the length of a cycle of interest as the cycle length threshold. To evaluate the significance of the observed cyclicity, the test statistic was compared against the maximum periodogram peak of the pseudo-series and p-values were computed. An α level of 0.05 was used as the threshold of significance.

3.3.2 Correlation between GISP2 and EDML

The correlation at various lags between GISP2 and EDML was evaluated using similar methods. After generating pseudo-series from GISP2 using the methods previously described, these series were compared to EDML and the maximum absolute value of the lagged correlations was computed, considering 27-year lags in the range of -2,500 to 2,500 years (Blunier et al. 1998; EPICA Community Members 2006). The inflated experiment-wise error rate associated with the simultaneous testing of multiple lagged correlations was corrected for by creating a distribution of the maximum absolute value of the lagged correlations in the spirit of Tukey's HSD (Higgins 2004). The correlation of each lag between the original records was then compared against this null distribution to determine significance. Also considered was the correlation between GISP2 and EDML at a lag of 0 using methods that did not correct for simultaneous testing. The results associated with this procedure should only be

	Cycle Length Threshold	Short Series (47 to 20 kyr BP)	Long Series (47 to 12.5 kyr BP)
NOM	2,500 years	0.229	0.411
	3,000 years	0.284	0.502
OM	2,500 years	0.116	0.273
	3,000 years	0.154	0.355

Table 3.1: The p-values after considering the short and long series for the non-overlapping D-O events model (NOM) and the overlapping D-O events model (OM). Under these specifications and at an α level of 0.05, there are no significant cyclicities associated with the appearance of D-O events.

considered if we are interested in the correlation of GISP2 and EDML at a lag of 0 and no other lags.

3.4 Results and Discussion

To evaluate the existence of cyclicities in the manifestation of D-O events, the D-O events in the short and long series were considered under the NOM and OM. The corresponding p-values are summarized in Table (3.1). It appears that the OM consistently results in lower p-values when compared to the NOM, indicating that cyclicity is more plausible under the OM for D-O events. However, there is no combination of D-O event model, cycle length threshold, and data series length that yields a statistically significant p-value. Evidence for cyclicity is strongest when using the OM, the cycle length threshold of 2,500 years, and the shorter series with a p-value of 0.116. That is, in this setting, the pseudo-series (which randomly distribute D-O events throughout the 47 to 20 kyr BP time period) yield evidence of cyclicity that is more compelling than the original data in 11.6% of the replications.

Also evaluated was the number of D-O events that could have occurred during the short series. Essentially, each D-O event can be considered a Bernoulli trial in which the event either appears or does not appear at a given time. D-O events have

been hypothesized to occur every 1,470 years during glacial periods (Grootes and Stuiver 1997). Based on this information, between 4 and 18 D-O events could have appeared in the short series with probability greater than 0. Under the NOM and OM, the number of D-O events in the pseudo-series was allowed to vary between 4 and 18. Even after varying the number of D-O events that could appear in the short series, at an α level of 0.05, there is no significant evidence of cyclicity in the appearance of D-O events, as summarized by Figure (3.1).

We explored the possibility that the duration of D-O events was a contributor to these events appearing somewhat cyclic in their manifestation. To test this hypothesis, a duration multiplier was applied to each D-O event in the short series to either shrink or expand the events. The scaled D-O events were randomized and cycles of less than 3,000 years were examined. Because the NOM does not allow D-O events to overlap, the duration multipliers considered under this model were $m_{NOM} = 0.25$, 0.50, 0.75, and 1.00; however, under the OM, the duration multipliers considered were $m_{OM} = 0.25$, 0.50, 0.75, 1.00, 1.25, 1.50, 1.75, and 2.00. Figure (3.2) confirms the suspicion that the observed D-O event durations are ideal for making this series appear cyclic. Because D-O events involve evaluated temperatures for a sustained period of time, the (possibly false) evidence of cyclicity in the time series will be affected by the duration of the events. These analyses indicate that any perception of cyclicity in D-O events is accentuated by event durations that are optimal for appearing cyclic—shorter D-O events with more filler or longer D-O events with less filler each yield less dramatic evidence of cyclicity.

D-O events have been suggested to occur every 1,470 years but may occasionally skip an appearance or two; that is to say, the duration between D-O events could occasionally be 3,000 or 4,500 years (Rahmstorf 2003). In order to assess this argument, we considered the periodograms of the pseudo-series differently than previously described. Instead of identifying the maximum peak of the periodograms, we found

	NOM	OM
Short Series	0.258	0.140
Long Series	0.483	0.356

Table 3.2: The p-values after accounting for the possibility of skipping manifestations of D-O events, where the short series is 47 to 20 kyr BP and the long series is 47 to 12.5 kyr BP. Two models were employed: the non-overlapping D-O events model (NOM) and the overlapping D-O events model (OM). Under this paradigm and at an α level of 0.05, there is not enough evidence to conclude that D-O events are cyclical in nature.

the sum of the two highest peaks within a certain distance of each other at different periodicities. The value that maximized this procedure was retained to represent the particular pseudo-series. The same was done to the observed series after randomizing the filler as previously discussed. This new test statistic was compared to the collection of maximum sums computed from the pseudo-series to evaluate if the cyclicity observed from this perspective was statistically significant. Under this scenario, both the short and long series were considered with the NOM and OM. Table (3.2) contains these p-values. Because these p-values do not exceed an α level of 0.05, there is not enough evidence to conclude D-O events are cyclical after accounting for the possibility of D-O events skipping an appearance.

3.4.1 Correlation between Greenland and Antarctica Records

The correlation between GISP2 and EDML from 47 to 20 kyr BP was evaluated at various lags, as shown in Figure (3.3). The x-axis of this plot represents the lag in years. A positive value means that GISP2 leads EDML, whereas a negative value indicates that EDML leads GISP2. It appears that the correlation between these two records is greatest when Antarctica leads Greenland.

The statistical significance of the correlation between GISP2 and EDML was assessed under the NOM and OM. Figure (3.4) shows the resulting p-values at vari-

ous lags under the NOM. The black line represents the p-values computed using an approach similar to Tukey's HSD and the black point is the p-value at a lag of 0 when we did not correct for simultaneous testing. The red dotted line represents an α level of 0.05. Any observations that fall below this line are significant. The plot similar to Figure (3.4) was examined for the OM and was almost identical to the plot included in this manuscript.

As Figure (3.4) demonstrates, under the NOM, the correlations between GISP2 and EDML are significant when EDML leads GISP2 by 382 to 682 years. Similarly, when the OM is evaluated, significant correlations occur when EDML leads GISP2 by 382 to 682 years. In each of these two scenarios, the maximum significant correlation between GISP2 and EDML is 0.455 and occurs when EDML leads GISP2 by 545 years. Thus, it appears that temperatures in Antarctica lead temperatures in Greenland by 545 years, roughly agreeing with an Antarctica lead of 1,000 to 2,500 years (Blunier et al. 1998).

The evidence for cyclicity in observed D-O events is not compelling enough to conclude that D-O events are cyclical, regardless of the D-O event model, cycle length threshold, and data series length imposed. This does not mean that D-O events are not cyclical; rather, there is a lack of evidence to prove they are cyclical. However, for there to be a cycle governing the appearance of D-O events, some external force would have to be involved that causes this regularity. As of yet, an external force that has the same time scale has not been identified. This, coupled with the lack of evidence for cyclicity in observed D-O events, may suggest that D-O events are not cyclical in manifestation. Non-cyclicity of D-O events implies that internal forces govern this climate variability and that the internal climate system is inherently unstable and sensitive.

With this research, we explored two definitions of D-O events. The first definition forces a cooling period between warming events, and thus D-O events are not

allowed to overlap (NOM), whereas the second definition allows multiple warming events to occur before the previous warmings return to baseline, allowing D-O events to overlap (OM). Understanding which of these two definitions represents the appearance of D-O events will aid our understanding of the mechanism forcing their manifestation. Specifically, if the NOM describes D-O events, then the mechanism driving their appearance will have long time scale changes, such as ice sheet, whereas the OM representation of D-O events would point to mechanisms that can reorganize quickly.

The length of the lags that resulted in significant correlations between Greenland and Antarctica suggest that temperature in the Northern and Southern Hemispheres are linked via oceanic teleconnection. In particular, because Antarctica leads Greenland, it appears that the temperature in the Southern Hemisphere changes first, and that change is communicated to the Northern Hemisphere through the ocean. Because of the long length of the lag, an atmospheric teleconnnection is unlikely (Blunier et al. 1998). Considering this South-North connection and the lack of evidence for cyclicity in the observed D-O events, small climate changes in the Southern Hemisphere may result in amplified changes in the Northern Hemisphere, as demonstrated through D-O events. The larger volume of land and ice in the Northern Hemisphere may cause the amplifications of climate variability in the North. The signal between hemispheres may be transferred through the ocean. In order to evaluate this possibility, further research in this area must be completed.

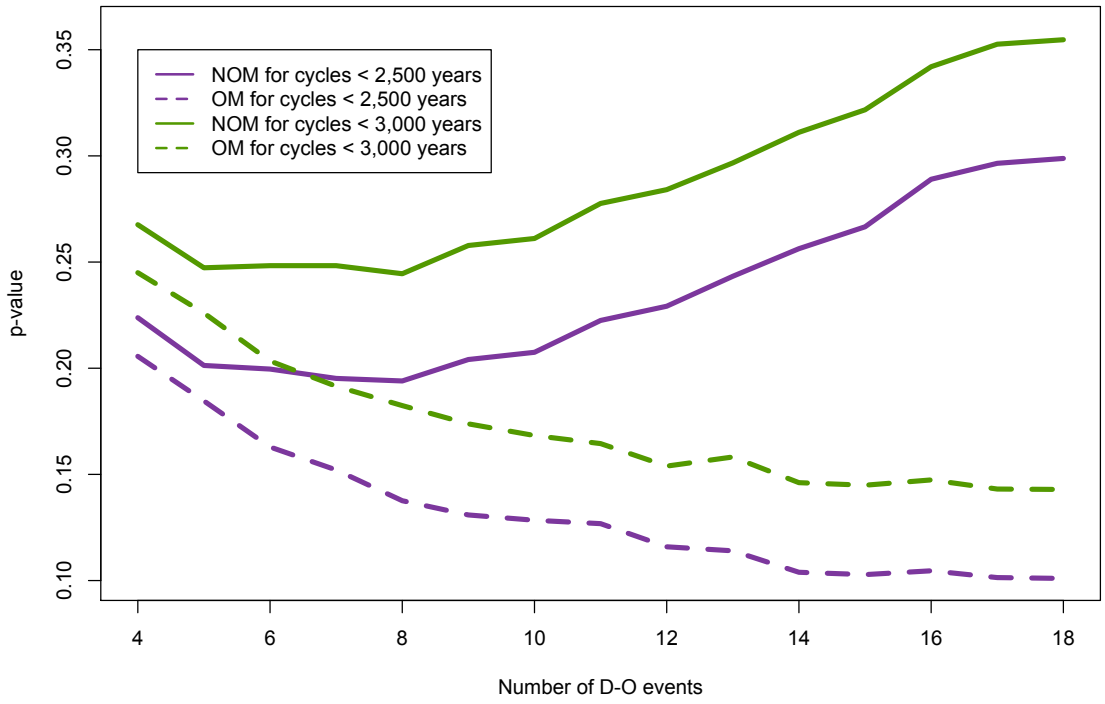


Figure 3.1: The p-values after allowing the number of D-O events in the series to vary from 4 to 18. Using the short series, the non-overlapping D-O events model (NOM) and overlapping D-O events model (OM) were considered. Regardless of the number of D-O events examined, at an α level of 0.05 there is not enough evidence to conclude that D-O events are cyclical in nature.

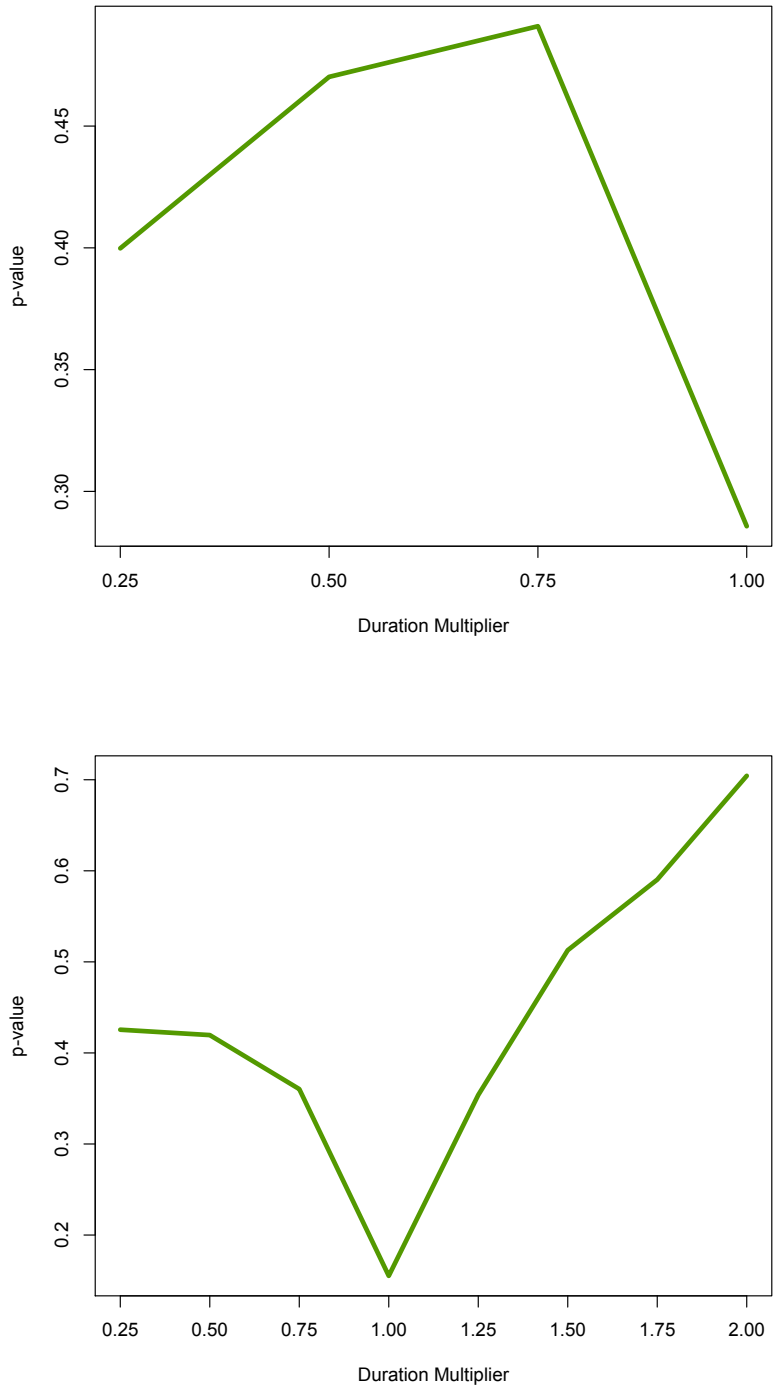


Figure 3.2: The p-values for the non-overlapping D-O events model (NOM, top plot) and the overlapping D-O events model (OM, bottom plot), where the duration multiplier is the amount the D-O events were retracted or expanded. The short series and cycles of less than 3,000 years were considered. The observed duration is ideal for this series appearing cyclic; however, at an α level of 0.05, there are no significant cyclicities.

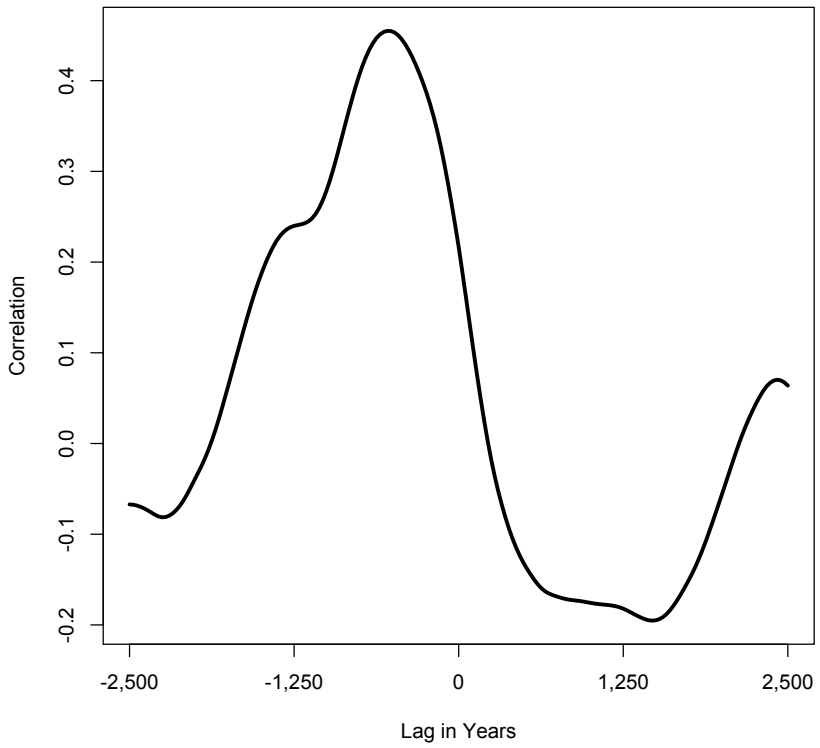


Figure 3.3: The correlation between the Greenland and Antarctica records at various lags. The records were considered from 47 to 20 kyr BP. A positive lag in years indicates that GISP2 leads EDML, and a negative lag in years means that EDML leads GISP2.

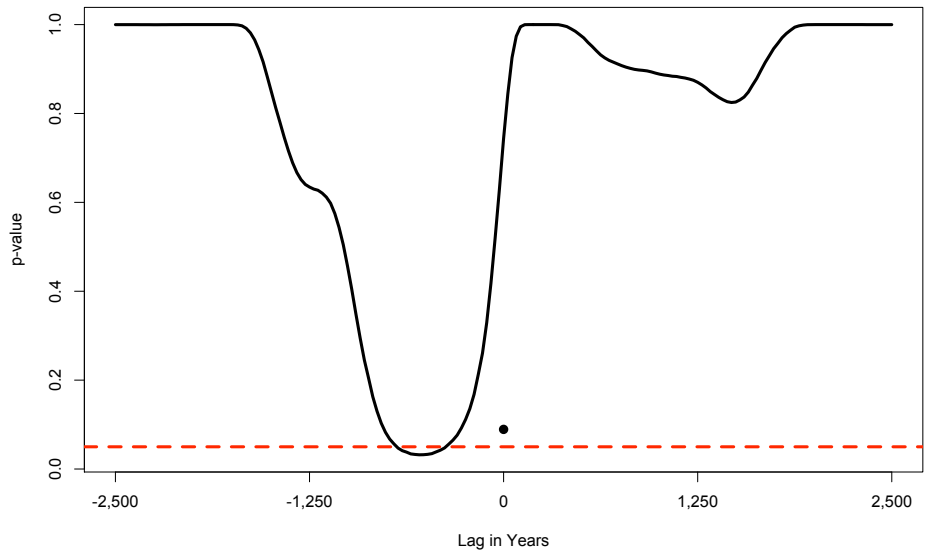


Figure 3.4: The black line represents the p-values, after correcting for simultaneous testing, associated with the correlation between GISP2 and EDML at each lag. The black point shows the p-value at a lag of 0 that has not been corrected for simultaneous tests, and the red dotted line represents an α level of 0.05. The non-overlapping D-O events model (NOM) was considered. When EDML leads GISP2 by 545 years, the correlation is 0.455, the maximum significant correlation.

4. CONCLUSIONS

The purpose of this thesis was twofold: to evaluate the evidence for cyclicity in observed D-O events and to assess whether or not the warming events observed in Greenland are related to the warming events observed in Antarctica. To accomplish both objectives, permutation tests and bootstrap approaches were employed. Specifically, the observations within the Greenland record were randomized by either shuffling or resampling D-O events. Statistics were accumulated from these new, randomized series in order to create null distributions that were used to assess the significance of what was observed. Two models were considered in these analyses: the non-overlapping D-O events model (NOM), which forces a cooling period between warming events, and the overlapping D-O events model (OM), which allows for a D-O event to occur before the previous D-O event has run its full course.

In particular, we found that there is not enough evidence to conclude that D-O events are cyclical in their manifestation. We considered the short and long series applied to the NOM and OM with shuffled D-O events and found the corresponding p-values to be non-significant at an α level of 0.05 for cycles less than 2,500 and 3,000 years. Likewise, after varying the number of D-O events that could appear in the short series, the p-values under the NOM and OM were also non-significant. We experimented with the duration of D-O events and found the observed length to be optimal for making D-O events appear cyclical in nature. That is, if the observed D-O events were any shorter or longer, the statistical evidence for cyclicity would be weaker still. Lastly, we allowed for the possibility of D-O events appearing approximately every 1,500 or 3,000 years. This analysis also yielded non-significant p-values. Thus, however one approaches this issue, there is not enough evidence to conclude that D-O events are cyclical in nature, regardless of what model is believed to be true.

To assess if the Greenland and Antarctica records were related, we considered the short series, shuffling and resampling D-O events using both the NOM and the OM. The correlation between the Greenland and the Antarctica record was computed for a variety of lags, the maximum lag corresponding to 2,500 years. Under each scenario, we found that a 545-year lagged correlation of 0.455 was significant at an α level of 0.05. This correlation occurs when the Antarctica record leads the Greenland record by 545 years and is the maximum significant correlation. When D-O events were shuffled, correlations with lags within the range of 327 to 736 years and 409 to 654 years were significant and all other lagged correlations were non-significant under the NOM and OM, respectively. Similarly, when considering D-O events that were resampled, correlations with lags within the range of 382 to 682 years were significant under both the NOM and the OM.

The evidence for cyclicity in observed D-O events is not compelling enough to conclude that D-O events are cyclical, regardless of the D-O event model, cycle length threshold, and data series length imposed. This does not mean that D-O events are not cyclical; rather, there is a lack of evidence to prove they are cyclical. However, for there to be a cycle governing the appearance of D-O events, some external force would have to be involved that causes this regularity. As of yet, an external force that has the same time scale has not been identified. This, coupled with the lack of evidence for cyclicity in observed D-O events, may suggest that D-O events are not cyclical in manifestation. Non-cyclicity of D-O events implies that internal forces govern this climate variability and that the internal climate system is inherently unstable and sensitive.

With this research, we explored two definitions of D-O events, the NOM and the OM. Understanding which of these two definitions represents the appearance of D-O events will aid our understanding of the mechanism forcing their manifestation. Specifically, if the NOM describes D-O events, then the mechanism driving their

appearance will have long time scale changes, such as ice sheet, whereas the OM representation of D-O events would point to mechanisms that can reorganize quickly.

The length of the lags that resulted in significant correlations between Greenland and Antarctica suggest that temperature in the Northern and Southern Hemispheres are linked via oceanic teleconnection. In particular, because Antarctica leads Greenland, it appears that the temperature in the Southern Hemisphere changes first, and that change is communicated to the Northern Hemisphere through the ocean. Because of the long length of the lag, an atmospheric teleconnnection is unlikely (Blunier et al. 1998). Considering this South-North connection and the lack of evidence for cyclicity in the observed D-O events, small climate changes in the Southern Hemisphere may result in amplified changes in the Northern Hemisphere, as demonstrated through D-O events, and this signal between hemispheres may be transferred through the ocean. In order to evaluate this possibility, further research in this area must be completed.

BIBLIOGRAPHY

Ackert Jr., R. P., Becker, R. A., Singer, B. S., Kurz, M. D., Caffee, M. W., and Mickelson, D. M. (2008), "Patagonian Glacier Response During the Late Glacial-Holocene Transition," *Science*, 321, 392–395.

Alley, R. B. and Clark, P. U. (1999), "The Deglaciation of the Northern Hemisphere: A Global Perspective," *Earth Planet Sciences*, 27, 149–182.

Blunier, T., Chappellez, J., Schwander, J., Dällenbach, A., Stauffer, B., Stocker, T. F., Raynaud, D., Jouzel, J., Clausen, H., Hammer, C., and Johnsen, S. J. (1998), "Asynchrony of Antarctic and Greenland climate change during the last glacial period," *Nature*, 394, 739–743.

Bond, G., Broecker, W., Johnsen, S., McManus, J., Labeyrie, L., Jouzel, J., and Bonani, G. (1993), "Correlations between climate records from North Atlantic sediments and Greenland ice," *Nature*, 365, 143–147.

Carlstein, E. (1986), "The Use of Subseries Values for Estimating the Variance of a General Statistic From a Stationary Sequence," *The Annals of Statistics*, 14, 1171–1179.

Christensen, W. F. and Sain, S. R. (2002), "Accounting for Dependence in a Flexible Multivariate Receptor Model," *Technometrics*, 44, 328–337.

Dansgaard, W., Johnsen, S. J., Clausen, H. B., Dahl-Jensen, D., Gundestrup, N. S., Hammer, C. U., Hvidberg, C. S., Steffensen, J. P., Sveinbjörnsdottir, A. E., Jouzel, J., and Bond, G. (1993), "Evidence for general instability of past climate from a 250-kyr ice-core record," *Nature*, 364, 218–219.

EPICA Community Members (2006), "One-to-one coupling of glacial climate variability in Greenland and Antarctica," *Nature*, 444, 195–198.

Grootes, P. M. and Stuiver, M. (1997), "Oxygen 18/16 variability in Greenland snow and ice with 10^{-3}- to 10^{-5}-year time resolution," *Journal of Geophysical Research*, 102, 26,455–26,470.

Hall, P. (1985), "Resampling a Coverage Process," *Stochastic Processes and Applications*, 20, 231–246.

Higgins, J. J. (2004), *Introduction to Modern Nonparametric Statistics*, Belmont, CA: Duxbury Press.

Huybers, P. and Curry, W. (2006), "Links between annual, Milankovitch and continuum temperature variability," *Nature*, 441, 329–332.

Jouzel, J., Masson-Delmotte, V., Cattani, O., Dreyfus, G., Falourd, S., Hoffmann, G., Minster, B., Nouet, J., Barnola, J. M., Chappellaz, J., Fischer, H., Gallet, J. C., Johnsen, S., Leuenberger, M., Loulergue, L., Luethi, D., Oerter, H., Parrenin, F., Raisbeck, G., Raynaud, D., Schilt, A., Schwander, J., Selmo, E., Souchez, R., Spahni, R., Stauffer, B., Steffensen, J. P., Stenni, B., Stocker, T. F., Tison, J. L., Werner, M., and Wolff, E. W. (2007), "Orbital and Millennial Antarctic Climate Variability over the Past 800,000 Years," *Science*, 317, 793–796.

Künsch, H. (1989), "The Jackknife and the Bootstrap for General Stationary Observations," *The Annals of Statistics*, 17, 1217–1241.

Lomb, N. R. (1976), "Least-Squares Frequency Analysis of Unequally Spaced Data," *Astrophysics and Space Science*, 39, 447–462.

Pardo-Igúzquiza, E. and Rodríguez-Tovar, F. J. (2000), "The permutation test as a non-parametric method for testing the statistical significance of power spectrum estimation in cyclostratigraphic research," *Earth and Planetary Sciences*, 181, 175–189.

Politis, D. N. and Romano, J. P. (1994), "The Stationary Bootstrap," *Journal of the American Statistical Association*, 89, 1303–1313.

Rahmstorf, S. (2003), "Timing of abrupt climate change: A precise clock," *Geophysical Research Letters*, 30, 1–4.

Roe, G. H. and Steig, E. J. (2004), "Characterization of Millennial-Scale Climate Variability," *Journal of Climate*, 17, 1929–1944.

Schulz, M. (2002), "On the 1470-year pacing of the Dansgaard-Oeschger warm events," *Paleoceanography*, 17, 1–9.

Steffensen, J. P., Andersen, K. K., Bigler, M., Clausen, H. B., Dahl-Jensen, D., Fischer, H., Goto-Azuma, K., Hansson, M., Johnsen, S. J., Jouzel, J., Masson-Delmotte, V., Popp, T., Rasmussen, S. O., Röthlisberger, R., Ruth, U., Stauffer, B., Siggaard-Andersen, M.-L., Árný E. Sveinbjörnsdóttir, Svensson, A., and White, J. W. C. (2008), "High-Resolution Greenland Ice Core Data Show Abrupt Climate Change Happens in Few Years," *Science*, 321, 680–683.

Steig, E. J. (2006), "The south-north connection," *Nature*, 444, 152–153.

Swanepoel, J. W. H. (1986), "The bootstrap applied to power spectral density function estimation," *Biometrika*, 73, 135–141.

A. STATISTICAL METHODS

The data employed for this thesis come from Greenland and Antarctica. Although many ice cores have been extracted from Greenland and Antarctica, this thesis uses one ice core from Greenland, GISP2, and one ice core from Antarctica, EDML. The GISP2 record is well-dated, making it a popular ice core to use in research (Roe and Steig, 2004). Likewise, EDML is widely considered a high-quality ice core when compared to other cores from Antarctica (EPICA Community Members, 2006). To prevent confusion, these ice core records will be referred to as the Greenland and Antarctica records. The data for Greenland and Antarctica are given in terms of years before present (BP) where present is considered to be 1950. Isotopes found in the ice cores act as surrogates for temperature: as temperature decreases, the isotopes under study become less abundant. The isotopes used for this research include δ^{18}O, the temperature metric for the Greenland record, and δD, used to represent the temperature from the Antarctica record. An initial examination of the Greenland and Antarctica records revealed a possible quadratic trend in the data. To eliminate this problem, both records were detrended and analyses were performed on the residuals of the detrended data.

Because of the nature of ice cores, the Greenland and Antarctica records contain unevenly spaced observations. Information is difficult to retrieve from ice cores and measurements are made wherever possible, regardless of the spacing between measurements. This presents problems when working with time series tools, such as periodograms. Another feature of these datasets is that the Greenland record is of a higher resolution than the Antarctica record. This means that there are more measured values and thus more observations for the same time frame in the Greenland record than in the Antarctica record. Again, this presents difficulties when comparing the Greenland and Antarctica records. Lomb (1976) proposes a method for

considering the power spectrum of unevenly spaced data. We did not use the procedure suggested by Lomb because the data also needed to be transformed to the same resolution. Instead, to remedy these issues, the data were linearly interpolated to contain 1,000 evenly spaced observations. Although this approach works well for the Greenland record, which contains 234 observations, the Antarctica record is extremely sparse, containing only 87 observations. This lack of data presents the problem of perhaps not truly representing the variability of temperature from Antarctica. For instance, more data may reveal a larger number of peaks or peaks of a higher magnitude in the Antarctica record. In spite of the short-comings associated with this sparse record, this version is the only one we were able to obtain and may be the highest resolution available at this time. Thus, although the results associated with a comparison of the Greenland and Antarctica records are interesting, they are not conclusive until a better, higher resolution of the Antarctica record can be obtained.

Since D-O events have indisputably occurred during the most recent glacial period, the window of consideration can be constricted to include this time frame only. Twelve D-O events, events 2 through 13, occurred between 47,245 to 20,000 year BP. This time frame will be referred to as the short series. The first D-O event occurs during the beginning of the transition between the glacial and interglacial periods. Because of its location, there is not a general consensus as to if this really is a D-O event. To be thorough, we considered the time frame that included this event as well, which ranges from 47,245 to 12,500 year BP. This window will be called the long series. The short and long series were used for the cyclicity analysis but only the short series was used when comparing the Greenland and Antarctica records. The reason for this is because the Antarctica record clearly moves into the transition between the glacial and interglacial period earlier than the Greenland record.

The occurrence of D-O events can be viewed from two perspectives: D-O events are either non-overlapping or overlapping in nature. Under the non-overlapping per-

spective, D-O events are formed by separate warming periods that can only begin after an earlier warming event has entirely concluded with temperatures returning to baseline. That is, a particular D-O event is considered one warming event and not a series of warming events that overlap one another. The overlapping view of D-O events allows for D-O events to be formed by a series of smaller, overlapping warming events. Both perspectives were explored through this research. For simplicity, the non-overlapping view is referred to as the non-overlapping D-O events model (NOM) and the overlapping perspective is referred to as the overlapping D-O events model (OM).

Under the OM, two positions can be taken: (1) D-O events 8 and 12 are not the result of overlapping D-O events but are simply longer than other D-O events, or (2) D-O events 8 and 12 are the result of overlapping D-O events and should not be considered in the data generation for the analysis. We explored both positions and found the conclusions from both views to be the same. For the purposes of this paper, we will discuss the findings of the first perspective when referencing the OM.

A.1 Cyclicity Detection

In order to assess the cyclicity of D-O events, permutation testing and bootstrapping methodologies were applied. Specifically, the D-O events within the observed series were randomly relocated in the time series to destroy any cyclicity that may have existed among the actual observed sequence of D-O events. We call this new series the pseudo-series because it consists solely of data found in the original series but reordered in such a way that D-O events are observable but not cyclical. The scale of each pseudo-series was adjusted so that the variance of each pseudo-series is equal to the variance of the original series. The cyclicity evident in each pseudo-series was quantified by recording the maximum periodogram peak, where peaks corresponding to cycles less than 2,500 and 3,000 years were considered. This

measurement of cyclicity was compared to the cyclicity found in the observed series. To evaluate the significance of the observed cycle, p-values were computed and an α level of 0.05 was used as the threshold of significance.

Because observations within the short series are dependent over time, it was essential to preserve the D-O events in this bootstrapping-type approach. Although the objective was to create series that were randomized, the structure and dependencies inherently associated with D-O events are important to preserve. Thus, instead of randomizing each observation, the D-O events themselves were randomized. D-O events were identified objectively using a piece-wise linear model algorithm. The observations between the D-O events, referred to as the "filler," also contain dependencies over time. Because of this, the filler was randomized such that groups of sequential observations would remain together in the new series. Specifically, a randomly selected value indicated where on the filler to start sampling from. Any filler before the randomly selected starting value was simply appended to the end of the filler observations. Consecutive pieces of filler values were then selected and randomly placed between the rearranged D-O events, preserving some dependencies natural to this time series. The resulting series shares certain important characteristics with the short series, such as preserving the actual D-O events and spanning the same amount of time, but applying this bootstrapping method allowed the resulting series to be random. These resulting series will be referred to as pseudo-series, which act as possible realizations of temperature records under the assumption of the hypothesis of no cyclicity.

Two approaches were taken to randomize the D-O events within the pseudo-series. The first approach consisted of shuffling the D-O events. This procedure implies that the D-O events were sampled without replacement and thus D-O events 2 through 13 were allowed to appear only once in the pseudo-series. By shuffling the D-O events, these 12 D-O events represent all the possible D-O events that could have

occurred, limiting the null hypothesis. Conversely, the second approach, resampling, samples the D-O events with replacement. The number of D-O events allowed in the pseudo-series was constrained to match that of the original, short series. That is, the pseudo-series was such that it could only contain 12 D-O events; however, the D-O events that appeared in the pseudo-series were randomly selected with replacement, allowing the possibility of some D-O events appearing more than once and others not at all in the pseudo-series. Under this procedure, these 12 D-O events represent a random sample of D-O events that could have occurred and leads to a broader null hypothesis. These two methods of randomizing the D-O events were completed under both the NOM and OM.

The periodogram of the pseudo-series was then examined to evaluate cyclicity. Periodograms are tools that display any possible cyclicity within a time series as peaks. The height of the peak indicates whether or not there is strong evidence of a cycle. Specifically, very short peaks are likely noise and tall peaks indicate strong evidence of cyclicity. From the periodogram, we collected the height of the tallest peak that corresponded to cycles less than 2,500, 3,000, and 5,000 years. Although D-O events have been suggested to occur approximately every 1,470 years, we considered these three cycles to be thorough (Grootes and Stuiver, 1997). Considering these higher-frequency peaks accounts for the possibility of D-O events having a different cyclicity due to noise or D-O events really having two cycles, one of about 1,500 years and another, less pronounced cycle at 3,000 years. This latter scenario could happen if a D-O event skips a manifestation but occurs 1,500 years after it failed to appear, as suggested by Rahmstorf (2003). Because the results from cycles less than 5,000 years provided no new or insightful information, we will only discuss cycles less than 2,500 and 3,000 years. We simulated this entire process using 10,000 repetitions and collected the maximum height for cycles less than 2,500 and 3,000 years each time. This collection of statistics formed two null distributions: a null distribution for cycles

less than 2,500 years and a null distribution for cycles less than 3,000 years.

In order to make inferences on the significance of the observed cyclicity, we created a test statistic to compare to the null distributions. It was necessary for the test statistic to represent the observed cyclicity due to the appearance of D-O events only. To capture this cycle, the filler was randomized to ensure that any cyclicity associated with it was removed. The filler was sampled using the same approach discussed with the creation of the pseudo-series. A periodogram of this series was produced and the maximum height pertaining to cycles less than 2,500 years and 3,000 years were both collected. Because removing the cyclicity associated with the filler introduces a random component, it was necessary to compute many test statistics through simulation. As before, 10,000 repetitions were used in this simulation. Each of these test statistics was compared to the appropriate null distribution and a p-value was computed, resulting in a p-value associated with each test statistic. Since we created 10,000 test statistics and thus had 10,000 p-values, we averaged the p-values such that there was only one p-value associated with each scenario of interest. For instance, we obtained an average p-value for the NOM when D-O events were shuffled and we considered cycles of less than 2,500 years. All results related to cyclicity are given in terms of averaged p-values. The same general method of building null distributions and test statistics was also implemented for the long series.

Several follow-up analyses were performed to better understand the observed cyclicity. These follow-up analyses consisted of the same procedure with some slight modifications. Specifically, when assessing the impact of varying the number of D-O events that could appear in a series, we considered the short series and the resampling randomization method. Likewise, when we considered the length, or duration, of a D-O event, we also used the short series and the resampling randomization method; however, we only considered cycles less than 3,000 years. A duration multiplier was applied to each D-O event to either expand or contract the D-O event before D-O

events were resampled. Lastly, we considered the possibility of D-O events having two cycles: a cycle of about 1,500 years and a cycle of a higher frequency for the instances when D-O events skip a manifestation. This required that a new null distribution be built based on a statistic that could incorporate two peaks in the periodogram. The statistic we retrieved was the sum of the two highest peaks in the periodogram that were within a certain distance of each other. This distance was double the given cycle on the periodogram with a buffer of 15% of that distance to account for any noise. To clarify, if we considered a cycle of 1,500 years, the distance metric we would compute would be $3{,}000 \pm 0.15 \times 3{,}000$. Because these distributions were built from different statistics, we also collected the appropriate test statistics to make comparisons under this paradigm.

A.2 Correlation between the Greenland and Antarctica Records

As mentioned previously, we considered the short series only when assessing the correlation between the Greenland and Antarctica records. The D-O events in the Greenland record were randomized using the shuffling and resampling methods and both the NOM and OM were considered. We evaluated the correlation between the Greenland and Antarctica record at various lags. We allowed the records to lag each other by 2,500 years because of the findings and suggestions of Blunier et al. (1998) and the EPICA Community Members (2006).

The same general procedure discussed in detail in the previous section was employed. After generating a pseudo-series of the Greenland record, this series was compared to the Antarctica record and two correlations were computed: the correlation at a lag of 0 and the maximum absolute value of the correlation, regardless of lag number. A simulation of 10,000 repetitions was performed and these statistics were gathered with each repetition. This collection of statistics resulted in two null distributions, a null distribution containing the correlation at a lag of 0 and a distri-

bution of the maximum absolute value of the correlation. The original Greenland and Antarctica records were assessed and the appropriate test statistics were retrieved to compare against the null distributions.

The p-values associated with the correlation at a lag of 0 do not take into account simultaneous tests. Because of this, the results should not be used unless we are only interested in describing the correlation between the Greenland and Antarctica records at a lag of 0. If we are instead interested in the correlation at other lags, the simultaneous testing must be accounted for to avoid inflating the experiment-wise error rate. In the spirit of Tukey's HSD (Higgins, 2004), we corrected for this inflation by creating a distribution of the maximum absolute value of the correlation. The correlation of each lag between the original records was then compared against this null distribution to determine significance.

B. ADDITIONAL RESULTS

B.1 Cyclicity of D-O events

The first analysis considered was shuffling the D-O events of both the short and long series under the NOM and OM. Table (B.1) summarizes the p-values from this analysis. It appears that the OM consistently results in lower p-values when compared to the NOM, indicating that cyclicity is more plausible under the OM for D-O events. However, there is no combination of D-O event model, cycle length threshold, and data series length that yields a statistically significant p-value. Evidence for cyclicity is strongest when using the OM, the cycle length threshold of 2,500 years, and the shorter series with a p-value of 0.06. That is, in this setting, the pseudo-series (which randomly distribute D-O events throughout the 47 to 20 kyr BP time period) yield evidence of cyclicity that is more compelling than the original data in 6% of the replications.

Because the first analysis showed that at an α level of 0.05 there is not enough evidence to prove that D-O events are cyclical in nature, regardless of the model, series, or amount of cyclicity under consideration, follow-up analyses were performed to better understand D-O events. The first follow-up analysis dealt with the number

	Cycle Length Threshold	Short Series (47 to 20 kyr BP)	Long Series (47 to 12.5 kyr BP)
NOM	2,500 years	0.14	0.17
	3,000 years	0.20	0.28
OM	2,500 years	0.06	0.09
	3,000 years	0.09	0.17

Table B.1: The p-values from shuffling the D-O events for the non-overlapping D-O events model (NOM) and the overlapping D-O events model (OM). Under these specifications and at an α level of 0.05, there are no significant cyclicities associated with the appearance of D-O events.

of D-O events that could have been observed in the short series. Essentially, each D-O event can be considered a Bernoulli trial in which the event either appears or does not appear at a given time. Using the notion that D-O events could occur approximately every 1,470 years (Schulz, 2002), new series were constructed to contain 4 to 18 D-O events. The probability of observing 4 to 18 D-O events is greater than 0, which is why as few as 4 D-O events were considered and as many as 18 D-O events were included. The resampling approach was applied to the short series under the NOM and OM and the p-values are summarized in Figure (B.1). It is again apparent that at an α level of 0.05 there is not enough evidence of cyclicity in D-O events, even when the number of D-O events is varied. As in Table (B.1), the OM results in lower p-values than the NOM when considering the same cycles. Likewise, cycles of less than 2,500 years result in lower p-values than cycles of less than 3,000 years. Note that these p-values based on resampling D-O events never look as significant as when we use the 12 actual D-O events. This indicates that these specific events accentuate the appearance of cyclicity more than sets of resampled D-O events.

The second follow-up analysis considered the length of D-O events. Specifically, the duration of D-O events could be a reason why these events appear somewhat cyclic in their manifestation, the length of D-O events forcing the cyclicity observed in the short and long series. To test this hypothesis, a duration multiplier was applied to each D-O event in the short series to either shrink or enlarge the events. The scaled D-O events were resampled and cycles of less than 3,000 years were examined. Because the NOM does not allow D-O events to overlap, the duration multipliers considered under this model were $m_{NOM} = 0.25$, 0.50, 0.75, and 1.00; however, under the OM, the duration multipliers considered were $m_{OM} = 0.25$, 0.50, 0.75, 1.00, 1.25, 1.50, 1.75, and 2.00. Figure (B.2) shows the p-values associated with this analysis where the top plot contains the p-values corresponding to the NOM and the bottom plot shows the results of the OM. Figure (B.2) confirms the suspicion that the duration of

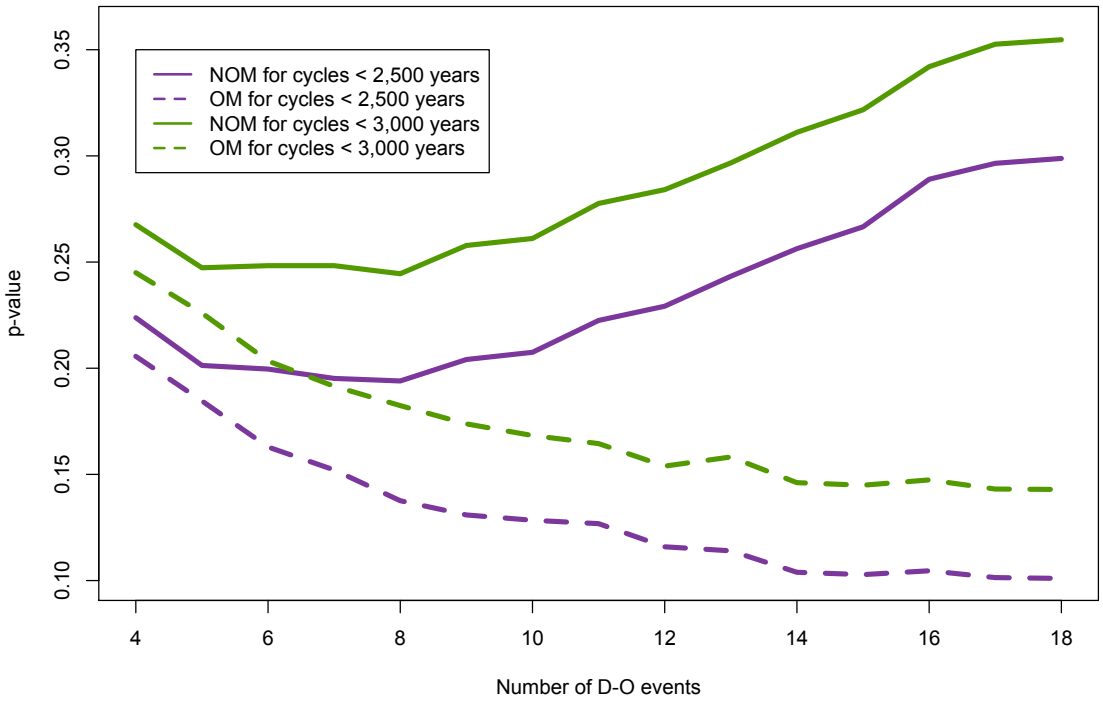


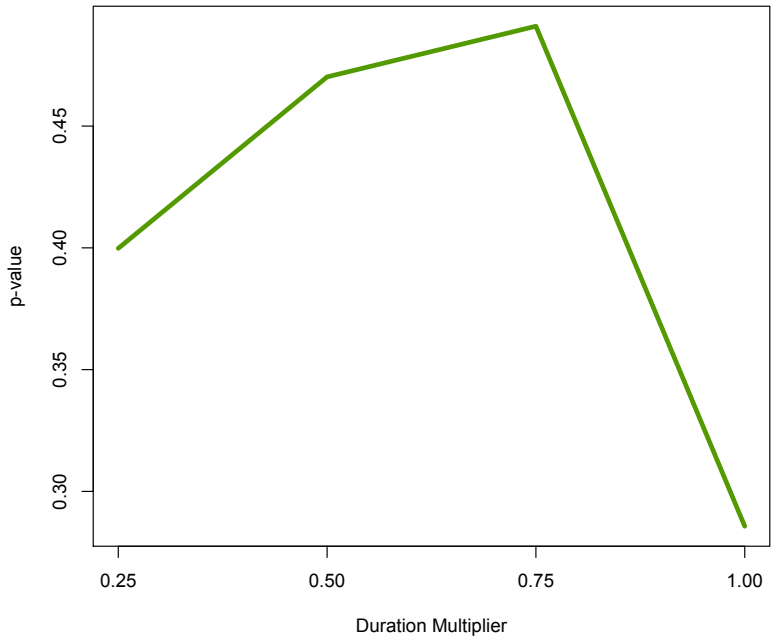
Figure B.1: The p-values corresponding to resampling D-O events, where the number of D-O events in the series was allowed to vary from 4 to 18. Using the short series, the non-overlapping D-O events model (NOM) and overlapping D-O events model (OM) were considered. Regardless of the number of D-O events examined, at an α level of 0.05 there is no significant cyclical component connected to D-O events.

	NOM	OM
Short Series	0.258	0.140
Long Series	0.483	0.356

Table B.2: The p-values after accounting for the possibility of skipping manifestations of D-O events, where the short series is 47 to 20 kyr BP and the long series is 47 to 12.5 kyr BP. Two models were employed: the non-overlapping D-O events model (NOM) and the overlapping D-O events model (OM). Under this paradigm and at an α level of 0.05, there is not enough evidence to conclude that D-O events are cyclical in nature.

a D-O event makes the series seem more or less cyclic, regardless of when the event actually appears in the series. In particular, the observed duration is ideal for making this series appear cyclical. Because D-O events involve evaluated temperatures for a sustained period of time, the (possibly false) evidence of cyclicity in the time series will be affected by the duration of the events. These analyses indicate that any perception of cyclicity in D-O events is accentuated by event durations that are optimal for appearing cyclic—shorter D-O events with more filler or longer D-O events with less filler each yield less dramatic evidence of cyclicity.

It has been suggested that D-O events occur approximately every 1,500 years but may occasionally skip a beat or two; that is to say, the duration between D-O events could occasionally be 3,000 or 4,500 years (Rahmstorf, 2003). This idea was assessed through a third follow-up analysis in which the statistic collected to build the null distribution and the test statistic was modified. Both the short series and the long series were considered under the NOM and OM. From Table (B.2), it appears that the p-values associated with the NOM and OM exceed an α level of 0.05, which means that there is not enough evidence to conclude that D-O events are cyclic in nature, even if D-O events skip a beat.



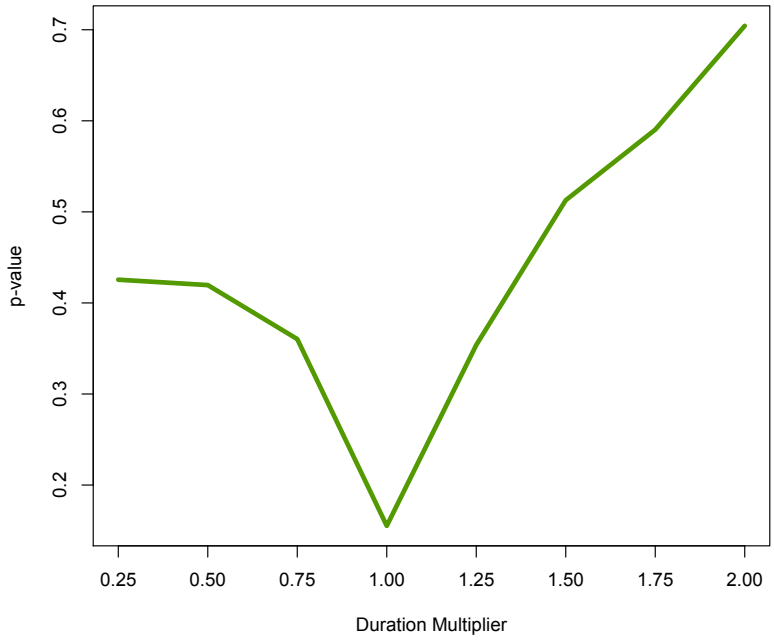


Figure B.2: The p-values for the non-overlapping D-O events model (NOM, top plot) and the overlapping D-O events model (OM, bottom plot), where the duration multiplier is the amount the D-O events were retracted or expanded. The short series and cycles of less than 3,000 years were considered. The observed duration is ideal for this series appearing cyclic; however, at an α level of 0.05, there are no significant cyclicities.

B.2 Correlation between the Greenland and Antarctica Records

The correlation between the Greenland and Antarctica records was found at various lags. The significance of these correlations was assessed using an approach similar to Tukey's HSD. In addition to this, the significance of the correlation between the Greenland and Antarctica records was evaluated at a lag of 0 using methods that did not correct for multiple comparisons. The records were compared using the short series and both the NOM and OM. Figure (B.3) shows the correlation between the Greenland and Antarctica records at various lags and Figure (B.4) shows the resulting p-values under the NOM and shuffled D-O events. Plots similar to Figure (B.4) were examined for each model and randomization method and were almost identical in appearance.

Figure (B.4) represents the p-values of each correlation under the NOM after shuffling the D-O events. The black line shows the p-values computed using an approach similar to Tukey's HSD. The black point represents the p-value at a lag of 0 using methods that do not account for multiple comparisons. This p-value should only be used if we are interested in the correlation at a lag of 0. The red dotted line represents an α level of 0.05 and any observations that fall below this line are significant. The x-axis in this plot is the lag in years. A positive value means that the Greenland record leads the Antarctica record and a negative value means the Antarctica record leads the Greenland record.

As Figure (B.4) shows, when the NOM is considered and D-O events are shuffled, the correlations between the Greenland and Antarctica records are significant when the Antarctica record leads the Greenland record by 327 to 736 years. When D-O events are again shuffled but the OM is evaluated, significant correlations occur when the Antarctica record leads the Greenland record by 409 to 654 years. In both cases, the maximum correlation between the Greenland and Antarctic records is 0.455 and happens when the Antarctica record leads the Greenland record by 545 years.

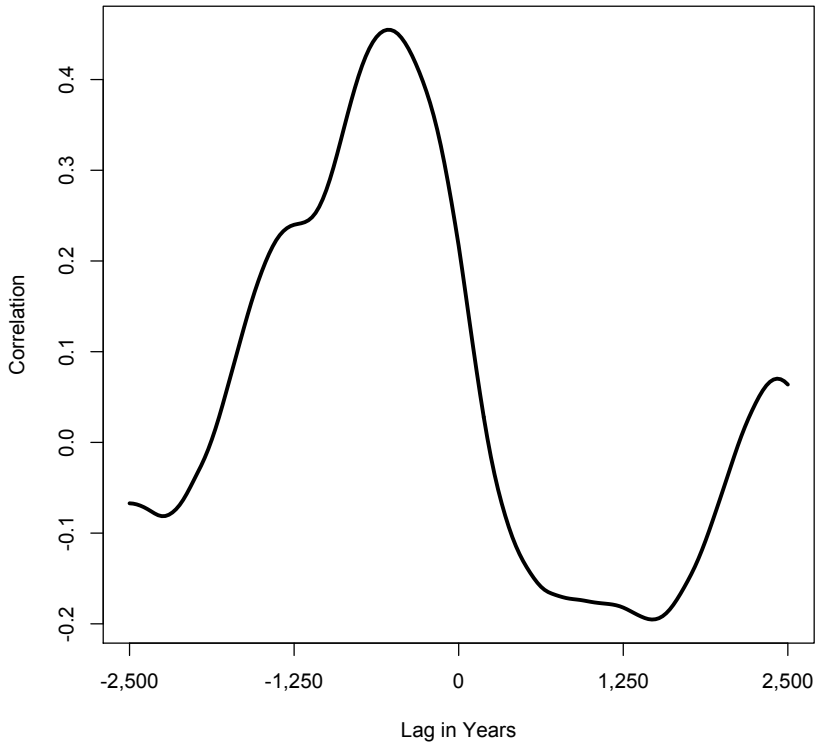


Figure B.3: The correlation between the Greenland and Antarctica records at various lags. The records were considered from 47 to 20 kyr BP. A positive lag in years indicates that the Greenland record leads the Antarctica record and a negative lag in years means that the Antarctica record leads the Greenland record.

Consider now the situation of resampling D-O events. For the NOM and OM, significant correlations between the two records occur when the Antarctica record leads the Greenland record by 382 to 682 years. As before, a correlation of 0.455, the maximum correlation between the Greenland and Antarctica records for both the NOM and OM, happens when the Antarctica record leads the Greenland record by 545 years. Thus, it appears that temperatures in Antarctica lead temperatures in Greenland by 545 years, roughly agreeing with the results of Blunier et al. (1998), which showed an Antarctica lead of 1,000 to 2,500 years.

Because there is a significant correlation between the Greenland and Antarctica records, we evaluated the connecting relationship. Specifically, we represented the Greenland record by $\tilde{x}(t) = \mu(t) + \epsilon(t)$ and the Antarctica record by $\tilde{y}(t) = \mu(t + \ell) + \xi(t)$ where t is time, ℓ is the lag, and $\mu(t)$ is the function that connects the temperatures in Greenland and Antarctica. To assess $\mu(t)$, we found the average between the Greenland and Antarctica records. Specifically, we scaled the detrended and linearly interpolated Greenland and Antarctica records to each have a variance of 1. The Antarctica record was shifted 545 years such that the correlation between the Greenland and shifted Antarctica record was 0.455, the maximum significant correlation, as shown in Figure (B.5). Using this shifted record of Antarctica, we found the average of the scaled residuals from the Greenland and Antarctica records for each available observation. This connecting function, $\mu(t)$, is summarized in Figure (B.6) and is a measure of the shape of this phenomena.

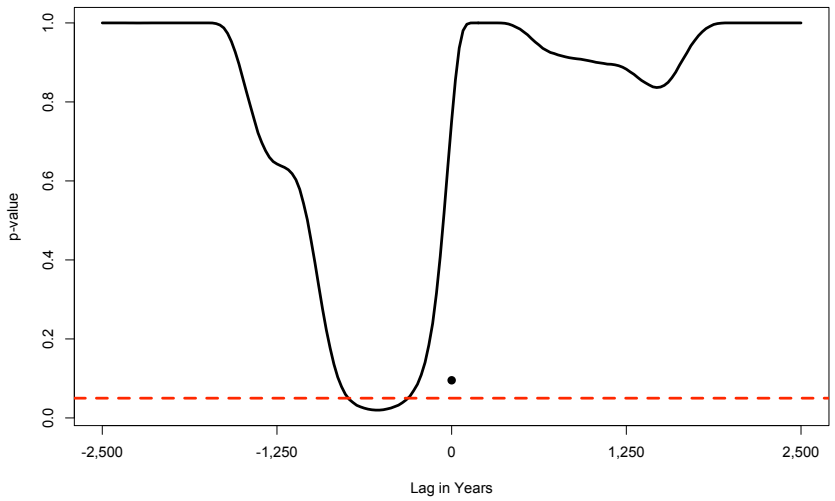


Figure B.4: The black line represents the p-values, after correcting for simultaneous testing, associated with the correlation between GISP2 and EDML at each lag. The black point shows the p-value at a lag of 0 that has not been corrected for simultaneous tests and the red dotted line represents an α level of 0.05. The non-overlapping D-O events model (NOM) was considered and the D-O events were shuffled. When EDML leads GISP2 by 545 years, the correlation is 0.455, the maximum significant correlation.

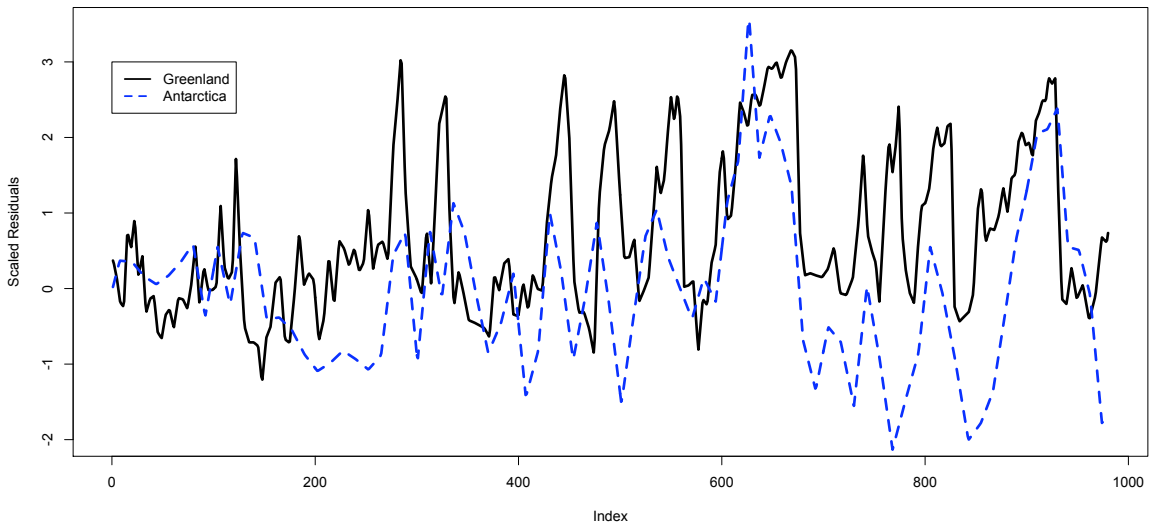


Figure B.5: The Greenland (black line) and Antarctica (blue dotted line) records after shifting the Antarctica record by 545 years such that the correlation between these two records is 0.455, the maximum significant correlation. The records are scaled such that each has a variance of 1.

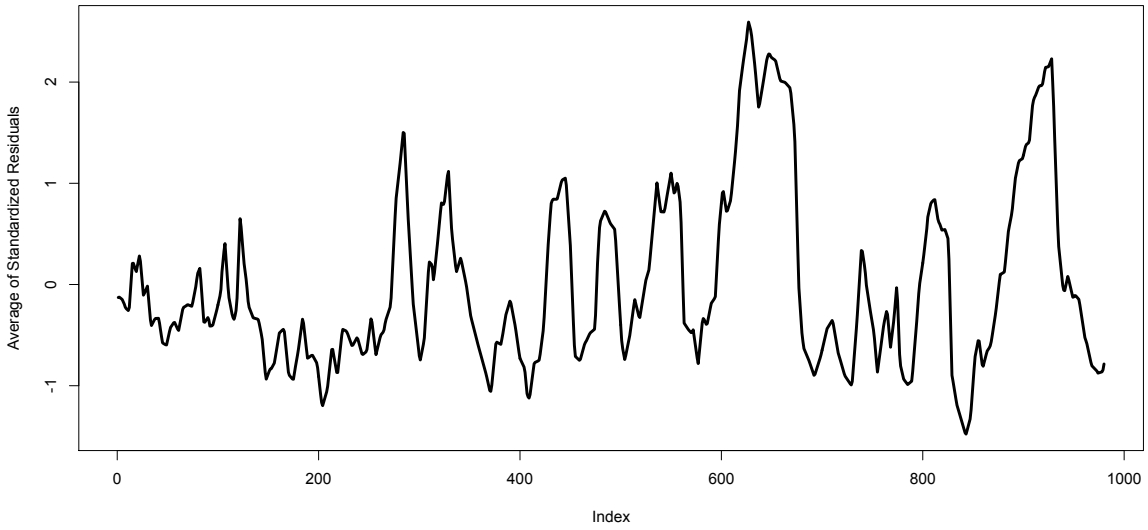


Figure B.6: The average between the Greenland and Antarctica record after shifting Antarctica by 545 years such that the correlation between these two records is 0.455. This represents the connecting function, $\mu(t)$, between the temperatures in Greenland and Antarctica.

C. SAMPLE OF CODE

A sample of the R and SAS code used for this thesis is shown in the sections below. Only a sample is included because much of the code is redundant with minor changes to adjust for the models and sampling methods. The code included here shows the analyses when considering the short series under the OM and shuffled D-O events. Note that GISP2 corresponds to the Greenland record and EDML is the Antarctica record.

C.1 Data Cleaning

```
# Detrending GISP2

gisp2new <- read.table("/Users/andrea/school/research/URGES/Thesis Project/Aspect 1/
Permutation Test Code/Window of Consideration/Simulation Datasets/GISP2 from 47 to 20
kyr BP.txt", header = TRUE)

# Retrieving D-O events
do2 <- gisp2new[33:34, ]
do3 <- gisp2new[65:69, ]
do4 <- gisp2new[75:80, ]
do5 <- gisp2new[98:105, ]
do6 <- gisp2new[109:117, ]
do7 <- gisp2new[119:130, ]
do8 <- gisp2new[134:157, ]
do9 <- gisp2new[167:169, ]
do10 <- gisp2new[172:179, ]
do11 <- gisp2new[182:193, ]
do12 <- gisp2new[194:220, ]
do13 <- gisp2new[227:233, ]

# Creating dataset of filler values with NAs where D-O events were

filler.detrend1 <- gisp2new
filler.detrend1[33:34, ] <- NA
filler.detrend1[65:69, ] <- NA
filler.detrend1[75:80, ] <- NA
filler.detrend1[98:105, ] <- NA
filler.detrend1[109:117, ] <- NA
filler.detrend1[119:130, ] <- NA
filler.detrend1[134:157, ] <- NA
filler.detrend1[167:169, ] <- NA
filler.detrend1[172:179, ] <- NA
filler.detrend1[182:193, ] <- NA
```

```
filler.detrend1[194:220, ] <- NA
filler.detrend1[227:233, ] <- NA

# Creating a quadratic model based on filler
mod1 <- lm(filler.detrend1[, 1] ~ filler.detrend1[, 2] + I(filler.detrend1[, 2]^2))

# Residuals of GISP2 (filler and D-O events) using the model above
all.fittedvals1 <- mod1$coeff[1] + mod1$coeff[2]*gisp2new[, 2] +
I(mod1$coeff[3]*(gisp2new[, 2]^2))

res1 <- gisp2new[, 1] - all.fittedvals1

# Checking the assumptions
plot(all.fittedvals1, res1, xlab = "Fitted", ylab = "Residuals")
qqnorm(res1)
qqline(res1)

detrend1 <- cbind(res1, gisp2new[, 2])

# Interpolating the detrended GISP2
detrend.int <- approx(detrend1[, 2], detrend1[, 1], xout = 20000)
end.obs <- cbind(detrend.int$y, detrend.int$x)
detrend.int1 <- detrend1[2:234, ]
detrend.int2 <- rbind(end.obs, detrend.int1)
detrend.int3 <- approx(detrend.int2[ , 2], detrend.int2[ , 1], n = 1000)
detrend.int4 <- cbind(detrend.int3$x, detrend.int3$y)

# Writing data to new file
write.table(detrend.int4, file = "/Users/andrea/school/research/URGES/Thesis
Project/Aspect 1/Permutation Test Code/Detrending Data/Simulation Datasets/GISP2
47-20kyr detrend interp.txt", row.names = FALSE, col.names = c("age", "res"))

# Detrending EDML
edml <- read.table("/Users/andrea/school/research/URGES/Thesis Project/Aspect 2/
Detrending Data/Simulation Datasets/edml.txt", header = TRUE)

# Creating quadratic model
mod <- lm(edml$deltaD ~ edml$age + I(edml$age^2))

# Computing the residuals based on this quadratic model
fitted.vals <- mod$coeff[1] + mod$coeff[2] * edml$age + I(mod$coeff[3] * (edml$age)^2)

res.edml <- edml$deltaD - fitted.vals

# Checking assumptions
plot(fitted.vals, res.edml, xlab = "Fitted", ylab = "Residuals")
qqnorm(res.edml)
qqline(res.edml)

detrend.edml <- cbind(res.edml, edml$age)

# Interpolating the detrended EDML
interp.edml <- approx(detrend.edml[, 2], detrend.edml[, 1], xout = c(20000, 47245))
end.obs <- cbind(interp.edml$y, interp.edml$x)
```

```
edml1 <- rbind(end.obs[1, ], detrend.edml[2:86, ], end.obs[2, ])
edml2 <- approx(edml1[, 2], edml1[, 1], n = 1000)
edml3 <- cbind(edml2$x, edml2$y)

# Writing data to new file
write.table(edml3, file = "/Users/andrea/school/research/URGES/Thesis Project/Aspect 2/
Data for Simulations/EDML 47-20 kyr detrend interp.txt", row.names = FALSE, col.names =
c("age", "res"))
```

C.2 Cyclicity Detection

C.2.1 D-O Event Identification

```
* Sample SAS code to identify where D-O event 2 starts and stop;
* This code was created by Alan Vaughn.

options formdlim="A";

data d1;
  input x y;
  cards;
   * imput data associated with general range of D-O event 2;
run;

data d2;
  set d1;
  newx = -x;
  drop x;
  rename newx=x;
run;

symbol i=join l=1 v=none;

proc gplot data=d2;
  plot y*x;
run;

proc nlin data=d2 method=dud;
  parms xo1=-22000 xo2=-21500 xo3=-21000 mu=-41 b1=0.02 b2=-.01;
  if x < xo1 then yhat=mu;
  if x >= xo1 and x < xo2 then yhat=mu + b1*(x-xo1);
  if x >= xo2 and x < xo3 then yhat=mu + b1*(xo2-xo1)+b2*(x-xo2);
  if x >= xo3 then yhat=mu + b1*(xo2-xo1)+b2*(xo3-xo2);
  model y=yhat;
  output out=d3 p=phat;
run;

proc gplot data=d3;
  plot phat*x;
run;

proc export data=d3
  outfile = '...\DO2 yhat.txt' dbms='tab';
```

```
run;
```

C.2.2 Null Distribution

```
# Read in the detrended and interpolated GISP2 from 47 to 20 kyr BP.
gisp2 <- read.table("/Users/andrea/school/research/URGES/Thesis Project/Aspect 1/
Permutation Test Code/Detrending Data/Simulation Datasets/GISP2 47-20kyr detrend
interp.txt", header = TRUE)

# Scaling GISP2
gisp2$res <- gisp2$res/sqrt(var(gisp2$res))

# This function is in preparation of the simulation. In it, we determine where
the filler starts and stops.
getstartstop <- function(j1)
{
  j1b <- c(j1[2:(length(j1))],-999999)
  j1f <- c(-999999,j1[1:(length(j1)-1)])
  starts <- is.na(j1) & !is.na(j1f)
  stops <- is.na(j1) & !is.na(j1b)
  #cbind(starts,stops)  # use this if you want an n by 2 matrix of T's and F's
  ## or ##
  cbind( (1:length(j1))[starts] , (1:length(j1))[stops] )
      # use this if you want a k by 2 matrix of start and stop locations
}

######################################################################
######################################################################
######################################################################
######################### Simulation #################################
######################################################################
######################################################################
######################################################################

all.sim.gmax <- NULL
all.sim.lmax <- NULL
DO.stat <- NULL
DO.stat.full <- NULL

numsim <- 10000

for(j in 1:numsim){

####################################################################
################### D-O Event Scattering#########################
####################################################################
# Creating 12 datasets with just 1 D-O event in each. The starting and
stopping values were specified by the piecewise linear model SAS code

do2 <- gisp2[119:129, 2]
do3 <- gisp2[272:292, 2]
do4 <- gisp2[316:337, 2]
do5 <- gisp2[423:455, 2]
do6 <- gisp2[475:516, 2]
```

```
do7 <- gisp2[521:563, 2]
do8 <- gisp2[581:677, 2]
do9 <- gisp2[731:744, 2]
do10 <- gisp2[755:777, 2]
do11 <- gisp2[786:829, 2]
do12 <- gisp2[834:933, 2]
do13 <- gisp2[966:998, 2]

filler <- gisp2[ , 2]
filler[119:129] <- NA
filler[272:292] <- NA
filler[316:337] <- NA
filler[423:455] <- NA
filler[475:516] <- NA
filler[521:563] <- NA
filler[581:677] <- NA
filler[731:744] <- NA
filler[755:777] <- NA
filler[786:829] <- NA
filler[834:933] <- NA
filler[966:998] <- NA

filler <- na.omit(filler)

donum <- 12
filler.num <- length(gisp2$age)

# Randomizing D-O events
do2.1 <- cbind(rand.nums[1], do2)
do3.1 <- cbind(rand.nums[2], do3)
do4.1 <- cbind(rand.nums[3], do4)
do5.1 <- cbind(rand.nums[4], do5)
do6.1 <- cbind(rand.nums[5], do6)
do7.1 <- cbind(rand.nums[6], do7)
do8.1 <- cbind(rand.nums[7], do8)
do9.1 <- cbind(rand.nums[8], do9)
do10.1 <- cbind(rand.nums[9], do10)
do11.1 <- cbind(rand.nums[10], do11)
do12.1 <- cbind(rand.nums[11], do12)
do13.1 <- cbind(rand.nums[12], do13)

rand.ordered <- rand.nums[order(rand.nums)]

obs.do2 <- which(ordered[ , 1] == rand.ordered[1])
obs.do3 <- which(ordered[ , 1] == rand.ordered[2])
obs.do4 <- which(ordered[ , 1] == rand.ordered[3])
obs.do5 <- which(ordered[ , 1] == rand.ordered[4])
obs.do6 <- which(ordered[ , 1] == rand.ordered[5])
obs.do7 <- which(ordered[ , 1] == rand.ordered[6])
obs.do8 <- which(ordered[ , 1] == rand.ordered[7])
obs.do9 <- which(ordered[ , 1] == rand.ordered[8])
obs.do10 <- which(ordered[ , 1] == rand.ordered[9])
obs.do11 <- which(ordered[ , 1] == rand.ordered[10])
obs.do12 <- which(ordered[ , 1] == rand.ordered[11])
```

```
obs.do13 <- which(ordered[ , 1] == rand.ordered[12])

new.do2 <- ordered[min(obs.do2):max(obs.do2), 2]
new.do3 <- ordered[min(obs.do3):max(obs.do3), 2]
new.do4 <- ordered[min(obs.do4):max(obs.do4), 2]
new.do5 <- ordered[min(obs.do5):max(obs.do5), 2]
new.do6 <- ordered[min(obs.do6):max(obs.do6), 2]
new.do7 <- ordered[min(obs.do7):max(obs.do7), 2]
new.do8 <- ordered[min(obs.do8):max(obs.do8), 2]
new.do9 <- ordered[min(obs.do9):max(obs.do9), 2]
new.do10 <- ordered[min(obs.do10):max(obs.do10), 2]
new.do11 <- ordered[min(obs.do11):max(obs.do11), 2]
new.do12 <- ordered[min(obs.do12):max(obs.do12), 2]
new.do13 <- ordered[min(obs.do13):max(obs.do13), 2]

rand1 <- rand.ordered[1]
rand2 <- rand.ordered[2]
rand3 <- rand.ordered[3]
rand4 <- rand.ordered[4]
rand5 <- rand.ordered[5]
rand6 <- rand.ordered[6]
rand7 <- rand.ordered[7]
rand8 <- rand.ordered[8]
rand9 <- rand.ordered[9]
rand10 <- rand.ordered[10]
rand11 <- rand.ordered[11]
rand12 <- rand.ordered[12]

diff1 <- rand2 - rand1
diff2 <- rand3 - rand2
diff3 <- rand4 - rand3
diff4 <- rand5 - rand4
diff5 <- rand6 - rand5
diff6 <- rand7 - rand6
diff7 <- rand8 - rand7
diff8 <- rand9 - rand8
diff9 <- rand10 - rand9
diff10 <- rand11 - rand10
diff11 <- rand12 - rand11

# The location in the empty dataset in which the first D-O event should be
inserted.
start1 <- rand1
end1 <- start1 + length(new.do2) - 1

start2 <- rand2
end2 <- start2 + length(new.do3) - 1

start3 <- rand3
end3 <- start3 + length(new.do4) - 1

start4 <- rand4
end4 <- start4 + length(new.do5) - 1
```

```
start5 <- rand5
end5 <- start5 + length(new.do6) - 1

start6 <- rand6
end6 <- start6 + length(new.do7) - 1

start7 <- rand7
end7 <- start7 + length(new.do8) - 1

start8 <- rand8
end8 <- start8 + length(new.do9) - 1

start9 <- rand9
end9 <- start9 + length(new.do10) - 1

start10 <- rand10
end10 <- start10 + length(new.do11) - 1

start11 <- rand11
end11 <- start11 + length(new.do12) - 1

start12 <- rand12
end12 <- start12 + length(new.do13) - 1


# Creating a dataset for each D-O event that contains just the one D-O
event and then values of -100 everywhere else. These values of -100 are dummy
values and we use them to hold the place of the D-O event. Each D-O event is
placed in its own dataset based on the randomly selected start value. We make
each dataset too large (we make it 2000 observations), so that if a D-O event
spills over 1000 observations it won't be truncated.

d1 <- c(rep(-100, (rand1 - 1)), new.do2, rep(-100, (2000 - (length(new.do2) +
rand1 - 1))))
d2 <- c(rep(-100, (rand2 - 1)), new.do3, rep(-100, (2000 - (length(new.do3) +
rand2 - 1))))
d3 <- c(rep(-100, (rand3 - 1)), new.do4, rep(-100, (2000 - (length(new.do4) +
rand3 - 1))))
d4 <- c(rep(-100, (rand4 - 1)), new.do5, rep(-100, (2000 - (length(new.do5) +
rand4 - 1))))
d5 <- c(rep(-100, (rand5 - 1)), new.do6, rep(-100, (2000 - (length(new.do6) +
rand5 - 1))))
d6 <- c(rep(-100, (rand6 - 1)), new.do7, rep(-100, (2000 - (length(new.do7) +
rand6 - 1))))
d7 <- c(rep(-100, (rand7 - 1)), new.do8, rep(-100, (2000 - (length(new.do8) +
rand7 - 1))))
d8 <- c(rep(-100, (rand8 - 1)), new.do9, rep(-100, (2000 - (length(new.do9) +
rand8 - 1))))
d9 <- c(rep(-100, (rand9 - 1)), new.do10, rep(-100, (2000 - (length(new.do10) +
rand9 - 1))))
d10 <- c(rep(-100, (rand10 - 1)), new.do11, rep(-100, (2000 - (length(new.do11)
+ rand10 - 1))))
d11 <- c(rep(-100, (rand11 - 1)), new.do12, rep(-100, (2000 - (length(new.do12)
+ rand11 - 1))))
```

```
d12 <- c(rep(-100, (rand12 - 1)), new.do13, rep(-100, (2000 - (length(new.do13)
+ rand12 - 1))))

# Now we create two datasets from each dataset above. We make sure we cut the
datasets at 1000, since this is the length we want test1 to be. Because we had
to go a little over 1000 to make sure we caught all of a D-O event that spills
over 1000, we decided to go to 2000 so that when we chop the datasets at 1000,
we will then have 2 datasets of equal length for each longer dataset. We did
this so that for pmax below.
d1.a <- d1[1:1000]
d1.b <- d1[1001:2000]
d2.a <- d2[1:1000]
d2.b <- d2[1001:2000]
d3.a <- d3[1:1000]
d3.b <- d3[1001:2000]
d4.a <- d4[1:1000]
d4.b <- d4[1001:2000]
d5.a <- d5[1:1000]
d5.b <- d5[1001:2000]
d6.a <- d6[1:1000]
d6.b <- d6[1001:2000]
d7.a <- d7[1:1000]
d7.b <- d7[1001:2000]
d8.a <- d8[1:1000]
d8.b <- d8[1001:2000]
d9.a <- d9[1:1000]
d9.b <- d9[1001:2000]
d10.a <- d10[1:1000]
d10.b <- d10[1001:2000]
d11.a <- d11[1:1000]
d11.b <- d11[1001:2000]
d12.a <- d12[1:1000]
d12.b <- d12[1001:2000]

# Now, using pmax, we can create one dataset that finds the max of the
vectors at each observation. We do this so that if we have any overlapping D-O
events, the max of the D-O events will be selected. We split the datasets in
half above so that if we had a D-O event spill over, it would then wrap to the
beginning, and then be considered for the max.
test1 <- pmax(d1.a, d1.b, d2.a, d2.b, d3.a, d3.b, d4.a, d4.b, d5.a, d5.b, d6.a,
d6.b, d7.a, d7.b, d8.a, d8.b, d9.a, d9.b, d10.a, d10.b, d11.a, d11.b, d12.a,
d12.b)

test1 <- ifelse(test1 == -100, NA, test1)

######################### Filler Points ##########################

# This step determines where to start sampling from in the filler data.
filler.rand <- sample(1:length(filler), 1, replace = FALSE)

filler.remain <- filler.rand - 1
if(filler.remain == 0){
filler.obs <- 0
} else
```

```
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per <- c(filler.per1, filler.per2)

filler.ss <- getstartstop(test1)

filler.na <- cbind(rep(NA, 20), rep(NA, 20))
filler.ss1 <- rbind(filler.ss, filler.na)
filler.ss2 <- filler.ss1[1:(donum + 1), ]

# The code below indicates where the D-O events are contained in the dataset
fstart1 <- filler.ss2[1, 1]
fend1 <- filler.ss2[1, 2]
flen1 <- fend1 - fstart1 + 1

flen1 <- ifelse(is.na(flen1) == TRUE, 0, flen1)

fstart2 <- filler.ss2[2, 1]
fend2 <- filler.ss2[2, 2]
flen2 <- fend2 - fstart2 + 1

flen2 <- ifelse(is.na(flen2) == TRUE, 0, flen2)

fstart3 <- filler.ss2[3, 1]
fend3 <- filler.ss2[3, 2]
flen3 <- fend3 - fstart3 + 1

flen3 <- ifelse(is.na(flen3) == TRUE, 0, flen3)

fstart4 <- filler.ss2[4, 1]
fend4 <- filler.ss2[4, 2]
flen4 <- fend4 - fstart4 + 1

flen4 <- ifelse(is.na(flen4) == TRUE, 0, flen4)

fstart5 <- filler.ss2[5, 1]
fend5 <- filler.ss2[5, 2]
flen5 <- fend5 - fstart5 + 1

flen5 <- ifelse(is.na(flen5) == TRUE, 0, flen5)

fstart6 <- filler.ss2[6, 1]
fend6 <- filler.ss2[6, 2]
flen6 <- fend6 - fstart6 + 1

flen6 <- ifelse(is.na(flen6) == TRUE, 0, flen6)

fstart7 <- filler.ss2[7, 1]
fend7 <- filler.ss2[7, 2]
flen7 <- fend7 - fstart7 + 1
```

```
flen7 <- ifelse(is.na(flen7) == TRUE, 0, flen7)

fstart8 <- filler.ss2[8, 1]
fend8 <- filler.ss2[8, 2]
flen8 <- fend8 - fstart8 + 1

flen8 <- ifelse(is.na(flen8) == TRUE, 0, flen8)

fstart9 <- filler.ss2[9, 1]
fend9 <- filler.ss2[9, 2]
flen9 <- fend9 - fstart9 + 1

flen9 <- ifelse(is.na(flen9) == TRUE, 0, flen9)

fstart10 <- filler.ss2[10, 1]
fend10 <- filler.ss2[10, 2]
flen10 <- fend10 - fstart10 + 1

flen10 <- ifelse(is.na(flen10) == TRUE, 0, flen10)

fstart11 <- filler.ss2[11, 1]
fend11 <- filler.ss2[11, 2]
flen11 <- fend11 - fstart11 + 1

flen11 <- ifelse(is.na(flen11) == TRUE, 0, flen11)

fstart12 <- filler.ss2[12, 1]
fend12 <- filler.ss2[12, 2]
flen12 <- fend12 - fstart12 + 1

flen12 <- ifelse(is.na(flen12) == TRUE, 0, flen12)

fstart13 <- filler.ss2[13, 1]
fend13 <- filler.ss2[13, 2]
flen13 <- fend13 - fstart13 + 1

flen13 <- ifelse(is.na(flen13) == TRUE, 0, flen13)

# Determine the order the filler points should be inserted.
rand.filler <- sample(1:(donum + 1), (donum + 1), replace = FALSE)

# Order the random numbers for the filler points.
frand.ordered <- rand.filler[order(rand.filler)]

flrand1 <- rep(frand.ordered[1], flen1)
flrand2 <- rep(frand.ordered[2], flen2)
flrand3 <- rep(frand.ordered[3], flen3)
flrand4 <- rep(frand.ordered[4], flen4)
flrand5 <- rep(frand.ordered[5], flen5)
flrand6 <- rep(frand.ordered[6], flen6)
flrand7 <- rep(frand.ordered[7], flen7)
flrand8 <- rep(frand.ordered[8], flen8)
flrand9 <- rep(frand.ordered[9], flen9)
```

```
flrand10 <- rep(frand.ordered[10], flen10)
flrand11 <- rep(frand.ordered[11], flen11)
flrand12 <- rep(frand.ordered[12], flen12)
flrand13 <- rep(frand.ordered[13], flen13)

flrand1 <- as.matrix(flrand1)
flrand2 <- as.matrix(flrand2)
flrand3 <- as.matrix(flrand3)
flrand4 <- as.matrix(flrand4)
flrand5 <- as.matrix(flrand5)
flrand6 <- as.matrix(flrand6)
flrand7 <- as.matrix(flrand7)
flrand8 <- as.matrix(flrand8)
flrand9 <- as.matrix(flrand9)
flrand10 <- as.matrix(flrand10)
flrand11 <- as.matrix(flrand11)
flrand12 <- as.matrix(flrand12)
flrand13 <- as.matrix(flrand13)

flrand <- rbind(flrand1, flrand2, flrand3, flrand4, flrand5, flrand6, flrand7,
flrand8, flrand9, flrand10, flrand11, flrand12, flrand13)

# Add the filler to the dataset randomly
obs.frand1 <- which(flrand == rand.filler[1])
obs.frand2 <- which(flrand == rand.filler[2])
obs.frand3 <- which(flrand == rand.filler[3])
obs.frand4 <- which(flrand == rand.filler[4])
obs.frand5 <- which(flrand == rand.filler[5])
obs.frand6 <- which(flrand == rand.filler[6])
obs.frand7 <- which(flrand == rand.filler[7])
obs.frand8 <- which(flrand == rand.filler[8])
obs.frand9 <- which(flrand == rand.filler[9])
obs.frand10 <- which(flrand == rand.filler[10])
obs.frand11 <- which(flrand == rand.filler[11])
obs.frand12 <- which(flrand == rand.filler[12])
obs.frand13 <- which(flrand == rand.filler[13])

if(length(obs.frand1) == 0){
obs.frand1 <- 0
}else
{
obs.frand1 <- min(obs.frand1):max(obs.frand1)
}

if(length(obs.frand2) == 0){
obs.frand2 <- 0
}else
{
obs.frand2 <- min(obs.frand2):max(obs.frand2)
}

if(length(obs.frand3) == 0){
obs.frand3 <- 0
}else
```

```
{
obs.frand3 <- min(obs.frand3):max(obs.frand3)
}

if(length(obs.frand4) == 0){
obs.frand4 <- 0
}else
{
obs.frand4 <- min(obs.frand4):max(obs.frand4)
}

if(length(obs.frand5) == 0){
obs.frand5 <- 0
}else
{
obs.frand5 <- min(obs.frand5):max(obs.frand5)
}

if(length(obs.frand6) == 0){
obs.frand6 <- 0
}else
{
obs.frand6 <- min(obs.frand6):max(obs.frand6)
}

if(length(obs.frand7) == 0){
obs.frand7 <- 0
}else
{
obs.frand7 <- min(obs.frand7):max(obs.frand7)
}

if(length(obs.frand8) == 0){
obs.frand8 <- 0
}else
{
obs.frand8 <- min(obs.frand8):max(obs.frand8)
}

if(length(obs.frand9) == 0){
obs.frand9 <- 0
}else
{
obs.frand9 <- min(obs.frand9):max(obs.frand9)
}

if(length(obs.frand10) == 0){
obs.frand10 <- 0
}else
{
obs.frand10 <- min(obs.frand10):max(obs.frand10)
}

if(length(obs.frand11) == 0){
```

```
obs.frand11 <- 0
}else
{
obs.frand11 <- min(obs.frand11):max(obs.frand11)
}

if(length(obs.frand12) == 0){
obs.frand12 <- 0
}else
{
obs.frand12 <- min(obs.frand12):max(obs.frand12)
}

if(length(obs.frand13) == 0){
obs.frand13 <- 0
}else
{
obs.frand13 <- min(obs.frand13):max(obs.frand13)
}

frand1 <- flrand[obs.frand1, ]
frand2 <- flrand[obs.frand2, ]
frand3 <- flrand[obs.frand3, ]
frand4 <- flrand[obs.frand4, ]
frand5 <- flrand[obs.frand5, ]
frand6 <- flrand[obs.frand6, ]
frand7 <- flrand[obs.frand7, ]
frand8 <- flrand[obs.frand8, ]
frand9 <- flrand[obs.frand9, ]
frand10 <- flrand[obs.frand10, ]
frand11 <- flrand[obs.frand11, ]
frand12 <- flrand[obs.frand12, ]
frand13 <- flrand[obs.frand13, ]

frand1 <- as.matrix(frand1)
frand2 <- as.matrix(frand2)
frand3 <- as.matrix(frand3)
frand4 <- as.matrix(frand4)
frand5 <- as.matrix(frand5)
frand6 <- as.matrix(frand6)
frand7 <- as.matrix(frand7)
frand8 <- as.matrix(frand8)
frand9 <- as.matrix(frand9)
frand10 <- as.matrix(frand10)
frand11 <- as.matrix(frand11)
frand12 <- as.matrix(frand12)
frand13 <- as.matrix(frand13)

frand <- rbind(frand1, frand2, frand3, frand4, frand5, frand6, frand7,
frand8, frand9, frand10, frand11, frand12, frand13)

c.frand <- cbind(frand, filler.per)

fordered <- c.frand[order(c.frand[, 1]), ]
```

```
obs.f1 <- which(fordered[ , 1] == frand.ordered[1])
obs.f2 <- which(fordered[ , 1] == frand.ordered[2])
obs.f3 <- which(fordered[ , 1] == frand.ordered[3])
obs.f4 <- which(fordered[ , 1] == frand.ordered[4])
obs.f5 <- which(fordered[ , 1] == frand.ordered[5])
obs.f6 <- which(fordered[ , 1] == frand.ordered[6])
obs.f7 <- which(fordered[ , 1] == frand.ordered[7])
obs.f8 <- which(fordered[ , 1] == frand.ordered[8])
obs.f9 <- which(fordered[ , 1] == frand.ordered[9])
obs.f10 <- which(fordered[ , 1] == frand.ordered[10])
obs.f11 <- which(fordered[ , 1] == frand.ordered[11])
obs.f12 <- which(fordered[ , 1] == frand.ordered[12])
obs.f13 <- which(fordered[ , 1] == frand.ordered[13])

if(length(obs.f1) == 0){
obs.f1 <- 0
}else
{
obs.f1 <- min(obs.f1):max(obs.f1)
}

if(length(obs.f2) == 0){
obs.f2 <- 0
}else
{
obs.f2 <- min(obs.f2):max(obs.f2)
}

if(length(obs.f3) == 0){
obs.f3 <- 0
}else
{
obs.f3 <- min(obs.f3):max(obs.f3)
}

if(length(obs.f4) == 0){
obs.f4 <- 0
}else
{
obs.f4 <- min(obs.f4):max(obs.f4)
}

if(length(obs.f5) == 0){
obs.f5 <- 0
}else
{
obs.f5 <- min(obs.f5):max(obs.f5)
}

if(length(obs.f6) == 0){
obs.f6 <- 0
}else
{
```

```
obs.f6 <- min(obs.f6):max(obs.f6)
}

if(length(obs.f7) == 0){
obs.f7 <- 0
}else
{
obs.f7 <- min(obs.f7):max(obs.f7)
}

if(length(obs.f8) == 0){
obs.f8 <- 0
}else
{
obs.f8 <- min(obs.f8):max(obs.f8)
}

if(length(obs.f9) == 0){
obs.f9 <- 0
}else
{
obs.f9 <- min(obs.f9):max(obs.f9)
}

if(length(obs.f10) == 0){
obs.f10 <- 0
}else
{
obs.f10 <- min(obs.f10):max(obs.f10)
}

if(length(obs.f11) == 0){
obs.f11 <- 0
}else
{
obs.f11 <- min(obs.f11):max(obs.f11)
}

if(length(obs.f12) == 0){
obs.f12 <- 0
}else
{
obs.f12 <- min(obs.f12):max(obs.f12)
}

if(length(obs.f13) == 0){
obs.frand13 <- 0
}else
{
obs.f13 <- min(obs.f13):max(obs.f13)
}


new.f1 <- fordered[obs.f1, 2]
```

```
new.f2 <- fordered[obs.f2, 2]
new.f3 <- fordered[obs.f3, 2]
new.f4 <- fordered[obs.f4, 2]
new.f5 <- fordered[obs.f5, 2]
new.f6 <- fordered[obs.f6, 2]
new.f7 <- fordered[obs.f7, 2]
new.f8 <- fordered[obs.f8, 2]
new.f9 <- fordered[obs.f9, 2]
new.f10 <- fordered[obs.f10, 2]
new.f11 <- fordered[obs.f11, 2]
new.f12 <- fordered[obs.f12, 2]
new.f13 <- fordered[obs.f13, 2]

if(length(new.f1) == 0){
f1.obs <- 0
}else
{
f1.obs <- fstart1:fend1
}

if(length(new.f2) == 0){
f2.obs <- 0
}else
{
f2.obs <- fstart2:fend2
}

if(length(new.f3) == 0){
f3.obs <- 0
}else
{
f3.obs <- fstart3:fend3
}

if(length(new.f4) == 0){
f4.obs <- 0
}else
{
f4.obs <- fstart4:fend4
}

if(length(new.f5) == 0){
f5.obs <- 0
}else
{
f5.obs <- fstart5:fend5
}

if(length(new.f6) == 0){
f6.obs <- 0
}else
{
f6.obs <- fstart6:fend6
}
```

```
if(length(new.f7) == 0){
f7.obs <- 0
}else
{
f7.obs <- fstart7:fend7
}

if(length(new.f8) == 0){
f8.obs <- 0
}else
{
f8.obs <- fstart8:fend8
}

if(length(new.f9) == 0){
f9.obs <- 0
}else
{
f9.obs <- fstart9:fend9
}

if(length(new.f10) == 0){
f10.obs <- 0
}else
{
f10.obs <- fstart10:fend10
}

if(length(new.f11) == 0){
f11.obs <- 0
}else
{
f11.obs <- fstart11:fend11
}

if(length(new.f12) == 0){
f12.obs <- 0
}else
{
f12.obs <- fstart12:fend12
}

if(length(new.f13) == 0){
f13.obs <- 0
}else
{
f13.obs <- fstart13:fend13
}

test1[f1.obs] <- new.f1
test1[f2.obs] <- new.f2
test1[f3.obs] <- new.f3
test1[f4.obs] <- new.f4
```

```
test1[f5.obs] <- new.f5
test1[f6.obs] <- new.f6
test1[f7.obs] <- new.f7
test1[f8.obs] <- new.f8
test1[f9.obs] <- new.f9
test1[f10.obs] <- new.f10
test1[f11.obs] <- new.f11
test1[f12.obs] <- new.f12
test1[f13.obs] <- new.f13

# Scaling test1
test1 <- (test1 - mean(test1))/(sd(test1))

################## End of D-O Event Scattering ##################



#####################################################################
########### Periodogram for Block Bootstrap of GISP2 ############
#####################################################################

sim.fs <- cbind(sim.p.gram$freq, sim.p.gram$spec)

# Finding the global maximum of spectrum
sim.max.loc <- which.max(sim.fs[ , 2])
sim.gmax <- sim.fs[sim.max.loc, ]
all.sim.gmax <- rbind(all.sim.gmax, sim.gmax)

# Finding local maximum
sim.fshort.loc1 <- which(sim.fs[ , 1] <= 0.1)
sim.fshort.loc2 <- max(sim.fshort.loc1)
sim.fs.short <- sim.fs[1:sim.fshort.loc2, ]

sim.spec <- as.matrix(sim.fs.short[ , 2])
sim.data1 <- rbind(0, sim.spec, 0) # Necessary to find a local max that may occur
at the end points
sim.data1 <- as.numeric(sim.data1)
sim.lmax.spec <- rep(NA, length(sim.data1))
for(i in 1:length(sim.data1)){
if(sim.data1[i] >= sim.data1[i+1] && sim.data1[i] >= sim.data1[i-1])
{sim.lmax.spec[i] = sim.data1[i]}
}

sim.loc.na <- is.na(sim.lmax.spec)
sim.non.na <- which(sim.loc.na == "FALSE")
sim.loc.non.na <- sim.non.na - 1

sim.lmax <- sim.fs.short[sim.loc.non.na, ]
all.sim.lmax <- rbind(all.sim.lmax, cbind(j, sim.lmax))

ordered <- all.sim.lmax[order(all.sim.lmax[ , 3], decreasing = TRUE), ]

######## End of Periodogram for Block Bootstrap of GISP2 ########
```

```
#####################################################################
###################### Statistic of Interest ########################
#####################################################################

order.freq <- sim.lmax[order(sim.lmax[, 1]), ]
DO.data <- matrix(order.freq, byrow = FALSE, ncol = 2)
DO.data <- cbind(j, DO.data)
DO.stat.full <- rbind(DO.stat.full, DO.data)

################## End of Statistic of Interest ################

}

####################### End of Simulation ######################

write.table(DO.stat.full, file = "/Users/andrea/school/research/URGES/Thesis
Project/Aspect 1/Permutation Test Code/Final Product/Simulations2/Results/
DOstat full for 47 to 20 kyr BP.txt", row.names = FALSE, col.names = c("simnum"
, "freq", "spec"))

#####################################################################
###### Future Analysis without having to run simulation #######
#####################################################################

dostat <- read.table("/Users/andrea/school/research/URGES/Thesis Project/Aspect
1/Permutation Test Code/Final Product/Simulations2/Results/DOstat full
for 47 to 20 kyr BP.txt", header = TRUE)

# Cycle less than 2500 (> 0.010908), 3000 (> 0.00909), and 5000 (> 0.005454)

DO.stat <- NULL
numsim <- 10000
for(i in 1:numsim){

obs.num <- which(dostat$simnum == i)

chunk <- dostat[min(obs.num):max(obs.num), ]

chunk <- chunk[, -1]

stat.wind <- which(chunk$freq > 0.05)
dostat.data <- chunk[min(stat.wind):max(stat.wind), ]
dostat.data <- as.matrix(dostat.data)

chunk.max <- which.max(dostat.data[, 2])

part.max <- dostat.data[chunk.max, ]
DO.stat <- rbind(DO.stat, part.max)

}

write.table(DO.stat, file = "/Users/andrea/school/research/URGES/Thesis Project/
```

```
Aspect 1/Permutation Test Code/Final Product/Simulations2/Results/DOstat part
for 47 to 20 kyr BP.txt", row.names = FALSE, col.names = c("freq", "spec"))
```

C.2.3 Test Statistic

```
gisp2 <- read.table(".../GISP2 47-20kyr detrend interp.txt", header = TRUE)

######################################################################
######################################################################
######################################################################
######################### Simulation #################################
######################################################################
######################################################################
######################################################################

all.sim.gmax <- NULL
all.sim.lmax <- NULL
DO.stat <- NULL
DO.stat.full <- NULL

numsim <- 10000

for(j in 1:numsim){

####################################################################
################### D-O Event Scattering#########################
####################################################################

do2 <- gisp2[119:129, 2]
do3 <- gisp2[272:292, 2]
do4 <- gisp2[316:337, 2]
do5 <- gisp2[423:455, 2]
do6 <- gisp2[475:516, 2]
do7 <- gisp2[521:563, 2]
do8 <- gisp2[581:677, 2]
do9 <- gisp2[731:744, 2]
do10 <- gisp2[755:777, 2]
do11 <- gisp2[786:829, 2]
do12 <- gisp2[834:933, 2]
do13 <- gisp2[966:998, 2]

filler <- gisp2[ , 2]
filler[119:129] <- NA
filler[272:292] <- NA
filler[316:337] <- NA
filler[423:455] <- NA
filler[475:516] <- NA
filler[521:563] <- NA
filler[581:677] <- NA
filler[731:744] <- NA
filler[755:777] <- NA
filler[786:829] <- NA
filler[834:933] <- NA
```

```
filler[966:998] <- NA

filler <- na.omit(filler)

donum <- 12

start1 <- 119
end1 <- start1 + length(do2) - 1

start2 <- 272
end2 <- start2 + length(do3) - 1

start3 <- 316
end3 <- start3 + length(do4) - 1

start4 <- 423
end4 <- start4 + length(do5) - 1

start5 <- 475
end5 <- start5 + length(do6) - 1

start6 <- 521
end6 <- start6 + length(do7) - 1

start7 <- 581
end7 <- start7 + length(do8) - 1

start8 <- 731
end8 <- start8 + length(do9) - 1

start9 <- 755
end9 <- start9 + length(do10) - 1

start10 <- 786
end10 <- start10 + length(do11) - 1

start11 <- 834
end11 <- start11 + length(do12) - 1

start12 <- 966
end12 <- start12 + length(do13) - 1

test1 <- rep(NA, length(gisp2$age))

test1[start1:end1] <- do2
test1[start2:end2] <- do3
test1[start3:end3] <- do4
test1[start4:end4] <- do5
test1[start5:end5] <- do6
test1[start6:end6] <- do7
test1[start7:end7] <- do8
test1[start8:end8] <- do9
test1[start9:end9] <- do10
test1[start10:end10] <- do11
```

```
test1[start11:end11] <- do12
test1[start12:end12] <- do13


######################### Filler Points ##########################

filler.rand <- sample(1:length(filler), 1, replace = FALSE)

filler.remain <- filler.rand - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per <- c(filler.per1, filler.per2)

fstart1 <- 1
fend1 <- start1 - 1
flen1 <- fend1 - fstart1 + 1

fstart2 <- end1 + 1
fend2 <- start2 - 1
flen2 <- fend2 - fstart2 + 1

fstart3 <- end2 + 1
fend3 <- start3 - 1
flen3 <- fend3 - fstart3 + 1

fstart4 <- end3 + 1
fend4 <- start4 - 1
flen4 <- fend4 - fstart4 + 1

fstart5 <- end4 + 1
fend5 <- start5 - 1
flen5 <- fend5 - fstart5 + 1

fstart6 <- end5 + 1
fend6 <- start6 - 1
flen6 <- fend6 - fstart6 + 1

fstart7 <- end6 + 1
fend7 <- start7 - 1
flen7 <- fend7 - fstart7 + 1

fstart8 <- end7 + 1
fend8 <- start8 - 1
flen8 <- fend8 - fstart8 + 1

fstart9 <- end8 + 1
```

```
fend9 <- start9 - 1
flen9 <- fend9 - fstart9 + 1

fstart10 <- end9 + 1
fend10 <- start10 - 1
flen10 <- fend10 - fstart10 + 1

fstart11 <- end10 + 1
fend11 <- start11 - 1
flen11 <- fend11 - fstart11 + 1

fstart12 <- end11 + 1
fend12 <- start12 - 1
flen12 <- fend12 - fstart12 + 1

fstart13 <- end12 + 1
fend13 <- length(test1)
flen13 <- fend13 - fstart13 + 1

rand.filler <- sample(1:(donum + 1), (donum + 1), replace = FALSE)

frand.ordered <- rand.filler[order(rand.filler)]

flrand1 <- rep(frand.ordered[1], flen1)
flrand2 <- rep(frand.ordered[2], flen2)
flrand3 <- rep(frand.ordered[3], flen3)
flrand4 <- rep(frand.ordered[4], flen4)
flrand5 <- rep(frand.ordered[5], flen5)
flrand6 <- rep(frand.ordered[6], flen6)
flrand7 <- rep(frand.ordered[7], flen7)
flrand8 <- rep(frand.ordered[8], flen8)
flrand9 <- rep(frand.ordered[9], flen9)
flrand10 <- rep(frand.ordered[10], flen10)
flrand11 <- rep(frand.ordered[11], flen11)
flrand12 <- rep(frand.ordered[12], flen12)
flrand13 <- rep(frand.ordered[13], flen13)

flrand1 <- as.matrix(flrand1)
flrand2 <- as.matrix(flrand2)
flrand3 <- as.matrix(flrand3)
flrand4 <- as.matrix(flrand4)
flrand5 <- as.matrix(flrand5)
flrand6 <- as.matrix(flrand6)
flrand7 <- as.matrix(flrand7)
flrand8 <- as.matrix(flrand8)
flrand9 <- as.matrix(flrand9)
flrand10 <- as.matrix(flrand10)
flrand11 <- as.matrix(flrand11)
flrand12 <- as.matrix(flrand12)
flrand13 <- as.matrix(flrand13)

flrand <- rbind(flrand1, flrand2, flrand3, flrand4, flrand5, flrand6, flrand7,
flrand8, flrand9, flrand10, flrand11, flrand12, flrand13)
```

```
obs.frand1 <- which(flrand == rand.filler[1])
obs.frand2 <- which(flrand == rand.filler[2])
obs.frand3 <- which(flrand == rand.filler[3])
obs.frand4 <- which(flrand == rand.filler[4])
obs.frand5 <- which(flrand == rand.filler[5])
obs.frand6 <- which(flrand == rand.filler[6])
obs.frand7 <- which(flrand == rand.filler[7])
obs.frand8 <- which(flrand == rand.filler[8])
obs.frand9 <- which(flrand == rand.filler[9])
obs.frand10 <- which(flrand == rand.filler[10])
obs.frand11 <- which(flrand == rand.filler[11])
obs.frand12 <- which(flrand == rand.filler[12])
obs.frand13 <- which(flrand == rand.filler[13])

if(length(obs.frand1) == 0){
obs.frand1 <- 0
}else
{
obs.frand1 <- min(obs.frand1):max(obs.frand1)
}

if(length(obs.frand2) == 0){
obs.frand2 <- 0
}else
{
obs.frand2 <- min(obs.frand2):max(obs.frand2)
}

if(length(obs.frand3) == 0){
obs.frand3 <- 0
}else
{
obs.frand3 <- min(obs.frand3):max(obs.frand3)
}

if(length(obs.frand4) == 0){
obs.frand4 <- 0
}else
{
obs.frand4 <- min(obs.frand4):max(obs.frand4)
}

if(length(obs.frand5) == 0){
obs.frand5 <- 0
}else
{
obs.frand5 <- min(obs.frand5):max(obs.frand5)
}

if(length(obs.frand6) == 0){
obs.frand6 <- 0
}else
{
obs.frand6 <- min(obs.frand6):max(obs.frand6)
```

```
}

if(length(obs.frand7) == 0){
obs.frand7 <- 0
}else
{
obs.frand7 <- min(obs.frand7):max(obs.frand7)
}

if(length(obs.frand8) == 0){
obs.frand8 <- 0
}else
{
obs.frand8 <- min(obs.frand8):max(obs.frand8)
}

if(length(obs.frand9) == 0){
obs.frand9 <- 0
}else
{
obs.frand9 <- min(obs.frand9):max(obs.frand9)
}

if(length(obs.frand10) == 0){
obs.frand10 <- 0
}else
{
obs.frand10 <- min(obs.frand10):max(obs.frand10)
}

if(length(obs.frand11) == 0){
obs.frand11 <- 0
}else
{
obs.frand11 <- min(obs.frand11):max(obs.frand11)
}

if(length(obs.frand12) == 0){
obs.frand12 <- 0
}else
{
obs.frand12 <- min(obs.frand12):max(obs.frand12)
}

if(length(obs.frand13) == 0){
obs.frand13 <- 0
}else
{
obs.frand13 <- min(obs.frand13):max(obs.frand13)
}

frand1 <- flrand[obs.frand1, ]
frand2 <- flrand[obs.frand2, ]
frand3 <- flrand[obs.frand3, ]
```

```
frand4 <- flrand[obs.frand4, ]
frand5 <- flrand[obs.frand5, ]
frand6 <- flrand[obs.frand6, ]
frand7 <- flrand[obs.frand7, ]
frand8 <- flrand[obs.frand8, ]
frand9 <- flrand[obs.frand9, ]
frand10 <- flrand[obs.frand10, ]
frand11 <- flrand[obs.frand11, ]
frand12 <- flrand[obs.frand12, ]
frand13 <- flrand[obs.frand13, ]

frand1 <- as.matrix(frand1)
frand2 <- as.matrix(frand2)
frand3 <- as.matrix(frand3)
frand4 <- as.matrix(frand4)
frand5 <- as.matrix(frand5)
frand6 <- as.matrix(frand6)
frand7 <- as.matrix(frand7)
frand8 <- as.matrix(frand8)
frand9 <- as.matrix(frand9)
frand10 <- as.matrix(frand10)
frand11 <- as.matrix(frand11)
frand12 <- as.matrix(frand12)
frand13 <- as.matrix(frand13)

frand <- rbind(frand1, frand2, frand3, frand4, frand5, frand6, frand7,frand8,
frand9, frand10, frand11, frand12, frand13)

c.frand <- cbind(frand, filler.per)

fordered <- c.frand[order(c.frand[, 1]), ]

obs.f1 <- which(fordered[ , 1] == frand.ordered[1])
obs.f2 <- which(fordered[ , 1] == frand.ordered[2])
obs.f3 <- which(fordered[ , 1] == frand.ordered[3])
obs.f4 <- which(fordered[ , 1] == frand.ordered[4])
obs.f5 <- which(fordered[ , 1] == frand.ordered[5])
obs.f6 <- which(fordered[ , 1] == frand.ordered[6])
obs.f7 <- which(fordered[ , 1] == frand.ordered[7])
obs.f8 <- which(fordered[ , 1] == frand.ordered[8])
obs.f9 <- which(fordered[ , 1] == frand.ordered[9])
obs.f10 <- which(fordered[ , 1] == frand.ordered[10])
obs.f11 <- which(fordered[ , 1] == frand.ordered[11])
obs.f12 <- which(fordered[ , 1] == frand.ordered[12])
obs.f13 <- which(fordered[ , 1] == frand.ordered[13])

if(length(obs.f1) == 0){
obs.f1 <- 0
}else
{
obs.f1 <- min(obs.f1):max(obs.f1)
}

if(length(obs.f13) == 0){
```

```
obs.frand13 <- 0
}else
{
obs.f13 <- min(obs.f13):max(obs.f13)
}


new.f1 <- fordered[obs.f1, 2]
new.f2 <- fordered[min(obs.f2):max(obs.f2), 2]
new.f3 <- fordered[min(obs.f3):max(obs.f3), 2]
new.f4 <- fordered[min(obs.f4):max(obs.f4), 2]
new.f5 <- fordered[min(obs.f5):max(obs.f5), 2]
new.f6 <- fordered[min(obs.f6):max(obs.f6), 2]
new.f7 <- fordered[min(obs.f7):max(obs.f7), 2]
new.f8 <- fordered[min(obs.f8):max(obs.f8), 2]
new.f9 <- fordered[min(obs.f9):max(obs.f9), 2]
new.f10 <- fordered[min(obs.f10):max(obs.f10), 2]
new.f11 <- fordered[min(obs.f11):max(obs.f11), 2]
new.f12 <- fordered[min(obs.f12):max(obs.f12), 2]
new.f13 <- fordered[obs.f13, 2]

if(length(new.f1) == 0){
f1.obs <- 0
}else
{
f1.obs <- fstart1:fend1
}

if(length(new.f13) == 0){
f13.obs <- 0
}else
{
f13.obs <- fstart13:fend13
}

test1[f1.obs] <- new.f1
test1[fstart2:fend2] <- new.f2
test1[fstart3:fend3] <- new.f3
test1[fstart4:fend4] <- new.f4
test1[fstart5:fend5] <- new.f5
test1[fstart6:fend6] <- new.f6
test1[fstart7:fend7] <- new.f7
test1[fstart8:fend8] <- new.f8
test1[fstart9:fend9] <- new.f9
test1[fstart10:fend10] <- new.f10
test1[fstart11:fend11] <- new.f11
test1[fstart12:fend12] <- new.f12
test1[f13.obs] <- new.f13

test1 <- na.omit(test1)
test1 <- test1/sqrt(var(test1))
################## End of D-O Event Scattering ##################

#################################################################
```

```
############ Periodogram for Block Bootstrap of GISP2 ############
####################################################################

sim.p.gram <- spec.pgram(test1, spans = c(2, 2), log = "no", taper = 0, plot =
FALSE)
sim.fs <- cbind(sim.p.gram$freq, sim.p.gram$spec)

sim.max.loc <- which.max(sim.fs[ , 2])
sim.gmax <- sim.fs[sim.max.loc, ]
all.sim.gmax <- rbind(all.sim.gmax, sim.gmax)

sim.fshort.loc1 <- which(sim.fs[ , 1] <= 0.1)
sim.fshort.loc2 <- max(sim.fshort.loc1)
sim.fs.short <- sim.fs[1:sim.fshort.loc2, ]

sim.spec <- as.matrix(sim.fs.short[ , 2])
sim.data1 <- rbind(0, sim.spec, 0)
sim.data1 <- as.numeric(sim.data1)
sim.lmax.spec <- rep(NA, length(sim.data1))
for(i in 1:length(sim.data1)){
if(sim.data1[i] >= sim.data1[i+1] && sim.data1[i] >= sim.data1[i-1])
{sim.lmax.spec[i] = sim.data1[i]}
}

sim.loc.na <- is.na(sim.lmax.spec)
sim.non.na <- which(sim.loc.na == "FALSE")
sim.loc.non.na <- sim.non.na - 1

sim.lmax <- sim.fs.short[sim.loc.non.na, ]
all.sim.lmax <- rbind(all.sim.lmax, cbind(j, sim.lmax))

ordered <- all.sim.lmax[order(all.sim.lmax[ , 3], decreasing = TRUE), ]

####### End of Periodogram for Block Bootstrap of GISP2 ########



####################################################################
#################### Statistic of Interest ####################
####################################################################

order.freq <- sim.lmax[order(sim.lmax[, 1]), ]
DO.data <- matrix(order.freq, byrow = FALSE, ncol = 2)
DO.data <- cbind(j, DO.data)
DO.stat.full <- rbind(DO.stat.full, DO.data)

################## End of Statistic of Interest ################

}

######################### End of Simulation ######################

write.table(DO.stat.full, file = "/Users/andrea/school/research/URGES/Thesis
Project/Aspect 1/Permutation Test Code/Final Product/Simulations2/Results/
```

```
DOstat full for 47 to 20 kyr BP.txt", row.names = FALSE, col.names = c("simnum",
"freq", "spec"))

################################################################
###### Future Analysis without having to run simulation #######
################################################################

dostat <- read.table(".../DOstat full
for 47 to 20 kyr BP.txt", header = TRUE)

# Cycle less than 2500 (> 0.010908), 3000 (> 0.00909), and 5000 (> 0.005454)

DO.stat <- NULL
numsim <- 10000
for(i in 1:numsim){

obs.num <- which(dostat$simnum == i)

chunk <- dostat[min(obs.num):max(obs.num), ]

chunk <- chunk[, -1]

stat.wind <- which(chunk$freq > 0.05)
dostat.data <- chunk[min(stat.wind):max(stat.wind), ]
dostat.data <- as.matrix(dostat.data)

chunk.max <- which.max(dostat.data[, 2])

part.max <- dostat.data[chunk.max, ]
DO.stat <- rbind(DO.stat, part.max)

}

write.table(DO.stat, file = "/Users/andrea/school/research/URGES/Thesis Project/
Aspect 1/Permutation Test Code/Final Product/Simulations2/Results/DOstat part
for 47 to 20 kyr BP.txt", row.names = FALSE, col.names = c("freq", "spec"))
```

C.2.4 Follow-Up Analyses

```
################################################################################
################################################################################
################################################################################
# Number of D-O events sample code - only 4 D-O events considered
################################################################################
################################################################################
################################################################################

gisp2 <- read.table("/Users/andrea/school/research/URGES/Thesis Project/Aspect 1
/Permutation Test Code/Detrending Data/Simulation Datasets/GISP2 47-20kyr
detrend interp.txt", header = TRUE)

# This function is in preparation of the simulation. In it, we figure out where
the filler starts and stops.
```

```
getstartstop <- function(j1)
{
  j1b <- c(j1[2:(length(j1))],-999999)
  j1f <- c(-999999,j1[1:(length(j1)-1)])
  starts <- is.na(j1) & !is.na(j1f)
  stops <- is.na(j1) & !is.na(j1b)
  #cbind(starts,stops)  # use this if you want an n by 2 matrix of T's and F's
  ## or ##
  cbind( (1:length(j1))[starts] , (1:length(j1))[stops] )
      # use this if you want a k by 2 matrix of start and stop locations
}

######################################################################
######################################################################
######################################################################
######################### Simulation #################################
######################################################################
######################################################################
######################################################################

all.sim.gmax <- NULL
all.sim.lmax <- NULL
DO.stat <- NULL
DO.stat.full <- NULL

numsim <- 10000

for(j in 1:numsim){

######################################################################
################### D-O Event Scattering#############################
######################################################################

do2 <- gisp2[119:129, 2]
do3 <- gisp2[272:292, 2]
do4 <- gisp2[316:337, 2]
do5 <- gisp2[423:455, 2]
do6 <- gisp2[475:516, 2]
do7 <- gisp2[521:563, 2]
do8 <- gisp2[581:677, 2]
do9 <- gisp2[731:744, 2]
do10 <- gisp2[755:777, 2]
do11 <- gisp2[786:829, 2]
do12 <- gisp2[834:933, 2]
do13 <- gisp2[966:998, 2]

filler <- gisp2[ , 2]
filler[119:129] <- NA
filler[272:292] <- NA
filler[316:337] <- NA
filler[423:455] <- NA
filler[475:516] <- NA
filler[521:563] <- NA
filler[581:677] <- NA
```

```
filler[731:744] <- NA
filler[755:777] <- NA
filler[786:829] <- NA
filler[834:933] <- NA
filler[966:998] <- NA

filler <- na.omit(filler)

rand.do <- sample(2:13, 4, replace = TRUE)

do2.rand <- cbind(do2, 2)
do3.rand <- cbind(do3, 3)
do4.rand <- cbind(do4, 4)
do5.rand <- cbind(do5, 5)
do6.rand <- cbind(do6, 6)
do7.rand <- cbind(do7, 7)
do8.rand <- cbind(do8, 8)
do9.rand <- cbind(do9, 9)
do10.rand <- cbind(do10, 10)
do11.rand <- cbind(do11, 11)
do12.rand <- cbind(do12, 12)
do13.rand <- cbind(do13, 13)

do.rand <- rbind(do2.rand, do3.rand, do4.rand, do5.rand, do6.rand, do7.rand,
do8.rand, do9.rand, do10.rand, do11.rand, do12.rand, do13.rand)

obs.do1.rand <- which(do.rand[ , 2] == rand.do[1])
obs.do2.rand <- which(do.rand[ , 2] == rand.do[2])
obs.do3.rand <- which(do.rand[ , 2] == rand.do[3])
obs.do4.rand <- which(do.rand[ , 2] == rand.do[4])

new.do1.rand <- do.rand[min(obs.do1.rand):max(obs.do1.rand), 1]
new.do2.rand <- do.rand[min(obs.do2.rand):max(obs.do2.rand), 1]
new.do3.rand <- do.rand[min(obs.do3.rand):max(obs.do3.rand), 1]
new.do4.rand <- do.rand[min(obs.do4.rand):max(obs.do4.rand), 1]

donum <- 4

do.obs <- sum(length(new.do1.rand), length(new.do2.rand), length(new.do3.rand),
length(new.do4.rand))

filler.num <- length(gisp2$age)
rand.nums <- sample(1:filler.num, donum, replace = FALSE)

do1.1 <- cbind(rand.nums[1], new.do1.rand)
do2.1 <- cbind(rand.nums[2], new.do2.rand)
do3.1 <- cbind(rand.nums[3], new.do3.rand)
do4.1 <- cbind(rand.nums[4], new.do4.rand)

doevents <- rbind(do1.1, do2.1, do3.1, do4.1)

ordered <- doevents[order(doevents[, 1]), ]
rand.ordered <- rand.nums[order(rand.nums)]
```

```
obs.do1 <- which(ordered[ , 1] == rand.ordered[1])
obs.do2 <- which(ordered[ , 1] == rand.ordered[2])
obs.do3 <- which(ordered[ , 1] == rand.ordered[3])
obs.do4 <- which(ordered[ , 1] == rand.ordered[4])

new.do1 <- ordered[min(obs.do1):max(obs.do1), 2]
new.do2 <- ordered[min(obs.do2):max(obs.do2), 2]
new.do3 <- ordered[min(obs.do3):max(obs.do3), 2]
new.do4 <- ordered[min(obs.do4):max(obs.do4), 2]

rand1 <- rand.ordered[1]
rand2 <- rand.ordered[2]
rand3 <- rand.ordered[3]
rand4 <- rand.ordered[4]

diff1 <- rand2 - rand1
diff2 <- rand3 - rand2
diff3 <- rand4 - rand3

start1 <- rand1
end1 <- start1 + length(new.do1) - 1

start2 <- rand2
end2 <- start2 + length(new.do2) - 1

start3 <- rand3
end3 <- start3 + length(new.do3) - 1

start4 <- rand4
end4 <- start4 + length(new.do4) - 1

d1 <- c(rep(-100, (rand1 - 1)), new.do1, rep(-100, (2000 - (length(new.do1) +
rand1 - 1))))
d2 <- c(rep(-100, (rand2 - 1)), new.do2, rep(-100, (2000 - (length(new.do2) +
rand2 - 1))))
d3 <- c(rep(-100, (rand3 - 1)), new.do3, rep(-100, (2000 - (length(new.do3) +
rand3 - 1))))
d4 <- c(rep(-100, (rand4 - 1)), new.do4, rep(-100, (2000 - (length(new.do4) +
rand4 - 1))))

d1.a <- d1[1:1000]
d1.b <- d1[1001:2000]
d2.a <- d2[1:1000]
d2.b <- d2[1001:2000]
d3.a <- d3[1:1000]
d3.b <- d3[1001:2000]
d4.a <- d4[1:1000]
d4.b <- d4[1001:2000]

test1 <- pmax(d1.a, d1.b, d2.a, d2.b, d3.a, d3.b, d4.a, d4.b)
test1 <- ifelse(test1 == -100, NA, test1)

######################### Filler Points ##########################
```

```
filler.rand <- sample(1:length(filler), (donum + 1), replace = TRUE)

filler.remain <- filler.rand[1] - 1
if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[1]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.1 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[2] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[2]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.2 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[3] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[3]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.3 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[4] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[4]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.4 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[5] - 1
```

```
if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[5]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.5 <- c(filler.per1, filler.per2)

filler.ss <- getstartstop(test1)
filler.na <- cbind(rep(NA, 20), rep(NA, 20))
filler.ss1 <- rbind(filler.ss, filler.na)
filler.ss2 <- filler.ss1[1:(donum + 1), ]

fstart1 <- filler.ss2[1, 1]
fend1 <- filler.ss2[1, 2]
flen1 <- fend1 - fstart1 + 1

flen1 <- ifelse(is.na(flen1) == TRUE, 0, flen1)

fstart2 <- filler.ss2[2, 1]
fend2 <- filler.ss2[2, 2]
flen2 <- fend2 - fstart2 + 1

flen2 <- ifelse(is.na(flen2) == TRUE, 0, flen2)

fstart3 <- filler.ss2[3, 1]
fend3 <- filler.ss2[3, 2]
flen3 <- fend3 - fstart3 + 1

flen3 <- ifelse(is.na(flen3) == TRUE, 0, flen3)

fstart4 <- filler.ss2[4, 1]
fend4 <- filler.ss2[4, 2]
flen4 <- fend4 - fstart4 + 1

flen4 <- ifelse(is.na(flen4) == TRUE, 0, flen4)

fstart5 <- filler.ss2[5, 1]
fend5 <- filler.ss2[5, 2]
flen5 <- fend5 - fstart5 + 1

flen5 <- ifelse(is.na(flen5) == TRUE, 0, flen5)

rand.filler <- sample(1:(donum + 1), (donum + 1), replace = FALSE)
frand.ordered <- rand.filler[order(rand.filler)]

flrand1 <- rep(frand.ordered[1], flen1)
flrand2 <- rep(frand.ordered[2], flen2)
flrand3 <- rep(frand.ordered[3], flen3)
flrand4 <- rep(frand.ordered[4], flen4)
```

```
flrand5 <- rep(frand.ordered[5], flen5)

flrand1 <- as.matrix(flrand1)
flrand2 <- as.matrix(flrand2)
flrand3 <- as.matrix(flrand3)
flrand4 <- as.matrix(flrand4)
flrand5 <- as.matrix(flrand5)

flrand <- rbind(flrand1, flrand2, flrand3, flrand4, flrand5)

obs.frand1 <- which(flrand == rand.filler[1])
obs.frand2 <- which(flrand == rand.filler[2])
obs.frand3 <- which(flrand == rand.filler[3])
obs.frand4 <- which(flrand == rand.filler[4])
obs.frand5 <- which(flrand == rand.filler[5])

if(length(obs.frand1) == 0){
obs.frand1 <- 0
}else
{
obs.frand1 <- min(obs.frand1):max(obs.frand1)
}

if(length(obs.frand2) == 0){
obs.frand2 <- 0
}else
{
obs.frand2 <- min(obs.frand2):max(obs.frand2)
}

if(length(obs.frand3) == 0){
obs.frand3 <- 0
}else
{
obs.frand3 <- min(obs.frand3):max(obs.frand3)
}

if(length(obs.frand4) == 0){
obs.frand4 <- 0
}else
{
obs.frand4 <- min(obs.frand4):max(obs.frand4)
}

if(length(obs.frand5) == 0){
obs.frand5 <- 0
}else
{
obs.frand5 <- min(obs.frand5):max(obs.frand5)
}

frand1 <- flrand[obs.frand1, ]
frand2 <- flrand[obs.frand2, ]
frand3 <- flrand[obs.frand3, ]
```

```
frand4 <- flrand[obs.frand4, ]
frand5 <- flrand[obs.frand5, ]

frand1 <- as.matrix(frand1)
frand2 <- as.matrix(frand2)
frand3 <- as.matrix(frand3)
frand4 <- as.matrix(frand4)
frand5 <- as.matrix(frand5)

frand <- rbind(frand1, frand2, frand3, frand4, frand5)

c.frand <- cbind(frand, filler.per.1, filler.per.2, filler.per.3, filler.per.4,
filler.per.5)

fordered <- c.frand[order(c.frand[, 1]), ]

obs.f1 <- which(fordered[ , 1] == frand.ordered[1])
obs.f2 <- which(fordered[ , 1] == frand.ordered[2])
obs.f3 <- which(fordered[ , 1] == frand.ordered[3])
obs.f4 <- which(fordered[ , 1] == frand.ordered[4])
obs.f5 <- which(fordered[ , 1] == frand.ordered[5])

if(length(obs.f1) == 0){
obs.f1 <- 0
}else
{
obs.f1 <- min(obs.f1):max(obs.f1)
}

if(length(obs.f2) == 0){
obs.f2 <- 0
}else
{
obs.f2 <- min(obs.f2):max(obs.f2)
}

if(length(obs.f3) == 0){
obs.f3 <- 0
}else
{
obs.f3 <- min(obs.f3):max(obs.f3)
}

if(length(obs.f4) == 0){
obs.f4 <- 0
}else
{
obs.f4 <- min(obs.f4):max(obs.f4)
}

if(length(obs.f5) == 0){
obs.frand5 <- 0
}else
{
```

```
obs.f5 <- min(obs.f5):max(obs.f5)
}


new.f1 <- fordered[obs.f1, 2]
new.f2 <- fordered[obs.f2, 3]
new.f3 <- fordered[obs.f3, 4]
new.f4 <- fordered[obs.f4, 5]
new.f5 <- fordered[obs.f5, 6]

if(length(new.f1) == 0){
f1.obs <- 0
}else
{
f1.obs <- fstart1:fend1
}

if(length(new.f2) == 0){
f2.obs <- 0
}else
{
f2.obs <- fstart2:fend2
}

if(length(new.f3) == 0){
f3.obs <- 0
}else
{
f3.obs <- fstart3:fend3
}

if(length(new.f4) == 0){
f4.obs <- 0
}else
{
f4.obs <- fstart4:fend4
}

if(length(new.f5) == 0){
f5.obs <- 0
}else
{
f5.obs <- fstart5:fend5
}

test1[f1.obs] <- new.f1
test1[f2.obs] <- new.f2
test1[f3.obs] <- new.f3
test1[f4.obs] <- new.f4
test1[f5.obs] <- new.f5

#Scaling test1
test1 <- test1/sqrt(var(test1))
```

```
################## End of D-O Event Scattering ##################



#####################################################################
########### Periodogram for Block Bootstrap of GISP2 ###########
#####################################################################

sim.p.gram <- spec.pgram(test1, spans = c(2, 2), log = "no", taper = 0, plot =
FALSE)
sim.fs <- cbind(sim.p.gram$freq, sim.p.gram$spec)

sim.max.loc <- which.max(sim.fs[ , 2])
sim.gmax <- sim.fs[sim.max.loc, ]
all.sim.gmax <- rbind(all.sim.gmax, sim.gmax)

sim.fshort.loc1 <- which(sim.fs[ , 1] <= 0.1)
sim.fshort.loc2 <- max(sim.fshort.loc1)
sim.fs.short <- sim.fs[1:sim.fshort.loc2, ]

sim.spec <- as.matrix(sim.fs.short[ , 2])
sim.data1 <- rbind(0, sim.spec, 0)
sim.data1 <- as.numeric(sim.data1)
sim.lmax.spec <- rep(NA, length(sim.data1))
for(i in 1:length(sim.data1)){
if(sim.data1[i] >= sim.data1[i+1] && sim.data1[i] >= sim.data1[i-1])
{sim.lmax.spec[i] = sim.data1[i]}
}

sim.loc.na <- is.na(sim.lmax.spec)
sim.non.na <- which(sim.loc.na == "FALSE")
sim.loc.non.na <- sim.non.na - 1

sim.lmax <- sim.fs.short[sim.loc.non.na, ]
all.sim.lmax <- rbind(all.sim.lmax, cbind(j, sim.lmax))
ordered <- all.sim.lmax[order(all.sim.lmax[ , 3], decreasing = TRUE), ]

####### End of Periodogram for Block Bootstrap of GISP2 ########



#####################################################################
#################### Statistic of Interest ####################
#####################################################################

order.freq <- sim.lmax[order(sim.lmax[, 1]), ]
DO.data <- matrix(order.freq, byrow = FALSE, ncol = 2)
DO.data <- cbind(j, DO.data)
DO.stat.full <- rbind(DO.stat.full, DO.data)

################## End of Statistic of Interest ################

}
```

```
######################## End of Simulation ######################

write.table(DO.stat.full, file = "Users/andrea/school/research/URGES/Thesis
Project/Aspect 1/Permutation Test Code/Final Product/Simulations2/Results/DOstat
full for 4 D-O events.txt", row.names = FALSE, col.names = c("simnum","freq",
"spec"))

#####################################################################
###### Future Analysis without having to run simulation #######
#####################################################################

dostat <- read.table("Users/andrea/school/research/URGES/Thesis Project/Aspect 1
/Permutation Test Code/Final Product/Simulations2/Results/DOstat full for 4 D-O
events.txt", header = TRUE)

# Want peaks less than 2500 (> 0.010908), 3000 (> 0.00909), and 5000 (> 0.005454)

DO.stat <- NULL
numsim <- 10000
for(i in 1:numsim){

obs.num <- which(dostat$simnum == i)

chunk <- dostat[min(obs.num):max(obs.num), ]

chunk <- chunk[, -1]

stat.wind <- which(chunk$freq > 0.005454)
dostat.data <- chunk[min(stat.wind):max(stat.wind), ]
dostat.data <- as.matrix(dostat.data)

chunk.max <- which.max(dostat.data[, 2])

part.max <- dostat.data[chunk.max, ]
DO.stat <- rbind(DO.stat, part.max)

}

write.table(DO.stat, file = "Users/andrea/school/research/URGES/Thesis
Project/Aspect 1/Permutation Test Code/Final
Product/Simulations2/Results/DOstat 5000 for 4 D-O events.txt", row.names =
FALSE, col.names = c("freq", "spec"))

#############################################################################
#############################################################################
#############################################################################
# Duration of D-O events
#############################################################################
#############################################################################
#############################################################################

gisp2 <- read.table(".../GISP2 47-20kyr detrend interp.txt", header = TRUE)
```

```
#####################################################################
###### Beginning of Expanding or Contracting D-O Events ########
#####################################################################

my.mult <- function(do2, do3, do4, do5, do6, do7, do8, do9, do10, do11, do12,
do13, m){

m2 <- length(do2$age) * m + 0.001
m3 <- length(do3$age) * m + 0.001
m4 <- length(do4$age) * m + 0.001
m5 <- length(do5$age) * m + 0.001
m6 <- length(do6$age) * m + 0.001
m7 <- length(do7$age) * m + 0.001
m8 <- length(do8$age) * m + 0.001
m9 <- length(do9$age) * m + 0.001
m10 <- length(do10$age) * m + 0.001
m11 <- length(do11$age) * m + 0.001
m12 <- length(do12$age) * m + 0.001
m13 <- length(do13$age) * m + 0.001

do2.m <- approx(do2$age, do2$res, n = round(m2, digits = 0))
do3.m <- approx(do3$age, do3$res, n = round(m3, digits = 0))
do4.m <- approx(do4$age, do4$res, n = round(m4, digits = 0))
do5.m <- approx(do5$age, do5$res, n = round(m5, digits = 0))
do6.m <- approx(do6$age, do6$res, n = round(m6, digits = 0))
do7.m <- approx(do7$age, do7$res, n = round(m7, digits = 0))
do8.m <- approx(do8$age, do8$res, n = round(m8, digits = 0))
do9.m <- approx(do9$age, do9$res, n = round(m9, digits = 0))
do10.m <- approx(do10$age, do10$res, n = round(m10, digits = 0))
do11.m <- approx(do11$age, do11$res, n = round(m11, digits = 0))
do12.m <- approx(do12$age, do12$res, n = round(m12, digits = 0))
do13.m <- approx(do13$age, do13$res, n = round(m13, digits = 0))

do2.m <- cbind(do2.m$y, 2)
do3.m <- cbind(do3.m$y, 3)
do4.m <- cbind(do4.m$y, 4)
do5.m <- cbind(do5.m$y, 5)
do6.m <- cbind(do6.m$y, 6)
do7.m <- cbind(do7.m$y, 7)
do8.m <- cbind(do8.m$y, 8)
do9.m <- cbind(do9.m$y, 9)
do10.m <- cbind(do10.m$y, 10)
do11.m <- cbind(do11.m$y, 11)
do12.m <- cbind(do12.m$y, 12)
do13.m <- cbind(do13.m$y, 13)

do.m <- rbind(do2.m, do3.m, do4.m, do5.m, do6.m, do7.m, do8.m,
do9.m, do10.m, do11.m, do12.m, do13.m)

do.m

}

do2.f <- gisp2[119:129, ]
```

```
do3.f <- gisp2[272:292, ]
do4.f <- gisp2[316:337, ]
do5.f <- gisp2[423:455, ]
do6.f <- gisp2[475:516, ]
do7.f <- gisp2[521:563, ]
do8.f <- gisp2[581:677, ]
do9.f <- gisp2[731:744, ]
do10.f <- gisp2[755:777, ]
do11.f <- gisp2[786:829, ]
do12.f <- gisp2[834:933, ]
do13.f <- gisp2[966:998, ]

m <- 0.25

################################################################

do.m <- my.mult(do2.f, do3.f, do4.f, do5.f, do6.f, do7.f, do8.f, do9.f, do10.f,
do11.f, do12.f, do13.f, m)

################################################################

do2.mobs <- which(do.m[, 2] == 2)
do3.mobs <- which(do.m[, 2] == 3)
do4.mobs <- which(do.m[, 2] == 4)
do5.mobs <- which(do.m[, 2] == 5)
do6.mobs <- which(do.m[, 2] == 6)
do7.mobs <- which(do.m[, 2] == 7)
do8.mobs <- which(do.m[, 2] == 8)
do9.mobs <- which(do.m[, 2] == 9)
do10.mobs <- which(do.m[, 2] == 10)
do11.mobs <- which(do.m[, 2] == 11)
do12.mobs <- which(do.m[, 2] == 12)
do13.mobs <- which(do.m[, 2] == 13)

do2.mnew <- do.m[min(do2.mobs):max(do2.mobs), 1]
do3.mnew <- do.m[min(do3.mobs):max(do3.mobs), 1]
do4.mnew <- do.m[min(do4.mobs):max(do4.mobs), 1]
do5.mnew <- do.m[min(do5.mobs):max(do5.mobs), 1]
do6.mnew <- do.m[min(do6.mobs):max(do6.mobs), 1]
do7.mnew <- do.m[min(do7.mobs):max(do7.mobs), 1]
do8.mnew <- do.m[min(do8.mobs):max(do8.mobs), 1]
do9.mnew <- do.m[min(do9.mobs):max(do9.mobs), 1]
do10.mnew <- do.m[min(do10.mobs):max(do10.mobs), 1]
do11.mnew <- do.m[min(do11.mobs):max(do11.mobs), 1]
do12.mnew <- do.m[min(do12.mobs):max(do12.mobs), 1]
do13.mnew <- do.m[min(do13.mobs):max(do13.mobs), 1]


########## End of Expanding or Contracting D-O Events ##########

getstartstop <- function(j1)
{
  j1b <- c(j1[2:(length(j1))],-999999)
  j1f <- c(-999999,j1[1:(length(j1)-1)])
```

```
  starts <- is.na(j1) & !is.na(j1f)
  stops <- is.na(j1) & !is.na(j1b)
  #cbind(starts,stops)  # use this if you want an n by 2 matrix of T's and F's
  ## or ##
  cbind( (1:length(j1))[starts] , (1:length(j1))[stops] )
      # use this if you want a k by 2 matrix of start and stop locations
}

#####################################################################
#####################################################################
#####################################################################
########################## Simulation ###############################
#####################################################################
#####################################################################
#####################################################################

all.sim.gmax <- NULL
all.sim.lmax <- NULL
DO.stat <- NULL
DO.stat.full <- NULL

numsim <- 10000

for(j in 1:numsim){

#####################################################################
################### D-O Event Scattering#############################
#####################################################################

do2 <- do2.mnew
do3 <- do3.mnew
do4 <- do4.mnew
do5 <- do5.mnew
do6 <- do6.mnew
do7 <- do7.mnew
do8 <- do8.mnew
do9 <- do9.mnew
do10 <- do10.mnew
do11 <- do11.mnew
do12 <- do12.mnew
do13 <- do13.mnew

filler <- gisp2[ , 2]
filler[119:129] <- NA
filler[272:292] <- NA
filler[316:337] <- NA
filler[423:455] <- NA
filler[475:516] <- NA
filler[521:563] <- NA
filler[581:677] <- NA
filler[731:744] <- NA
filler[755:777] <- NA
filler[786:829] <- NA
filler[834:933] <- NA
```

```
filler[966:998] <- NA

filler <- na.omit(filler)

rand.do <- sample(2:13, 12, replace = TRUE)

do2.rand <- cbind(do2, 2)
do3.rand <- cbind(do3, 3)
do4.rand <- cbind(do4, 4)
do5.rand <- cbind(do5, 5)
do6.rand <- cbind(do6, 6)
do7.rand <- cbind(do7, 7)
do8.rand <- cbind(do8, 8)
do9.rand <- cbind(do9, 9)
do10.rand <- cbind(do10, 10)
do11.rand <- cbind(do11, 11)
do12.rand <- cbind(do12, 12)
do13.rand <- cbind(do13, 13)

do.rand <- rbind(do2.rand, do3.rand, do4.rand, do5.rand, do6.rand, do7.rand,
do8.rand, do9.rand, do10.rand, do11.rand, do12.rand, do13.rand)

obs.do1.rand <- which(do.rand[ , 2] == rand.do[1])
obs.do2.rand <- which(do.rand[ , 2] == rand.do[2])
obs.do3.rand <- which(do.rand[ , 2] == rand.do[3])
obs.do4.rand <- which(do.rand[ , 2] == rand.do[4])
obs.do5.rand <- which(do.rand[ , 2] == rand.do[5])
obs.do6.rand <- which(do.rand[ , 2] == rand.do[6])
obs.do7.rand <- which(do.rand[ , 2] == rand.do[7])
obs.do8.rand <- which(do.rand[ , 2] == rand.do[8])
obs.do9.rand <- which(do.rand[ , 2] == rand.do[9])
obs.do10.rand <- which(do.rand[ , 2] == rand.do[10])
obs.do11.rand <- which(do.rand[ , 2] == rand.do[11])
obs.do12.rand <- which(do.rand[ , 2] == rand.do[12])

new.do1.rand <- do.rand[min(obs.do1.rand):max(obs.do1.rand), 1]
new.do2.rand <- do.rand[min(obs.do2.rand):max(obs.do2.rand), 1]
new.do3.rand <- do.rand[min(obs.do3.rand):max(obs.do3.rand), 1]
new.do4.rand <- do.rand[min(obs.do4.rand):max(obs.do4.rand), 1]
new.do5.rand <- do.rand[min(obs.do5.rand):max(obs.do5.rand), 1]
new.do6.rand <- do.rand[min(obs.do6.rand):max(obs.do6.rand), 1]
new.do7.rand <- do.rand[min(obs.do7.rand):max(obs.do7.rand), 1]
new.do8.rand <- do.rand[min(obs.do8.rand):max(obs.do8.rand), 1]
new.do9.rand <- do.rand[min(obs.do9.rand):max(obs.do9.rand), 1]
new.do10.rand <- do.rand[min(obs.do10.rand):max(obs.do10.rand), 1]
new.do11.rand <- do.rand[min(obs.do11.rand):max(obs.do11.rand), 1]
new.do12.rand <- do.rand[min(obs.do12.rand):max(obs.do12.rand), 1]

donum <- 12

do.obs <- sum(length(new.do1.rand), length(new.do2.rand), length(new.do3.rand),
length(new.do4.rand), length(new.do5.rand), length(new.do6.rand),
length(new.do7.rand), length(new.do8.rand), length(new.do9.rand),
length(new.do10.rand), length(new.do11.rand), length(new.do12.rand))
```

```
filler.num <- length(gisp2$age)

do1.1 <- cbind(rand.nums[1], new.do1.rand)
do2.1 <- cbind(rand.nums[2], new.do2.rand)
do3.1 <- cbind(rand.nums[3], new.do3.rand)
do4.1 <- cbind(rand.nums[4], new.do4.rand)
do5.1 <- cbind(rand.nums[5], new.do5.rand)
do6.1 <- cbind(rand.nums[6], new.do6.rand)
do7.1 <- cbind(rand.nums[7], new.do7.rand)
do8.1 <- cbind(rand.nums[8], new.do8.rand)
do9.1 <- cbind(rand.nums[9], new.do9.rand)
do10.1 <- cbind(rand.nums[10], new.do10.rand)
do11.1 <- cbind(rand.nums[11], new.do11.rand)
do12.1 <- cbind(rand.nums[12], new.do12.rand)

doevents <- rbind(do1.1, do2.1, do3.1, do4.1, do5.1, do6.1, do7.1,do8.1, do9.1,
do10.1, do11.1, do12.1)

ordered <- doevents[order(doevents[, 1]), ]
rand.ordered <- rand.nums[order(rand.nums)]

obs.do1 <- which(ordered[ , 1] == rand.ordered[1])
obs.do2 <- which(ordered[ , 1] == rand.ordered[2])
obs.do3 <- which(ordered[ , 1] == rand.ordered[3])
obs.do4 <- which(ordered[ , 1] == rand.ordered[4])
obs.do5 <- which(ordered[ , 1] == rand.ordered[5])
obs.do6 <- which(ordered[ , 1] == rand.ordered[6])
obs.do7 <- which(ordered[ , 1] == rand.ordered[7])
obs.do8 <- which(ordered[ , 1] == rand.ordered[8])
obs.do9 <- which(ordered[ , 1] == rand.ordered[9])
obs.do10 <- which(ordered[ , 1] == rand.ordered[10])
obs.do11 <- which(ordered[ , 1] == rand.ordered[11])
obs.do12 <- which(ordered[ , 1] == rand.ordered[12])

new.do1 <- ordered[min(obs.do1):max(obs.do1), 2]
new.do2 <- ordered[min(obs.do2):max(obs.do2), 2]
new.do3 <- ordered[min(obs.do3):max(obs.do3), 2]
new.do4 <- ordered[min(obs.do4):max(obs.do4), 2]
new.do5 <- ordered[min(obs.do5):max(obs.do5), 2]
new.do6 <- ordered[min(obs.do6):max(obs.do6), 2]
new.do7 <- ordered[min(obs.do7):max(obs.do7), 2]
new.do8 <- ordered[min(obs.do8):max(obs.do8), 2]
new.do9 <- ordered[min(obs.do9):max(obs.do9), 2]
new.do10 <- ordered[min(obs.do10):max(obs.do10), 2]
new.do11 <- ordered[min(obs.do11):max(obs.do11), 2]
new.do12 <- ordered[min(obs.do12):max(obs.do12), 2]

rand1 <- rand.ordered[1]
rand2 <- rand.ordered[2]
rand3 <- rand.ordered[3]
rand4 <- rand.ordered[4]
rand5 <- rand.ordered[5]
rand6 <- rand.ordered[6]
```

```
rand7 <- rand.ordered[7]
rand8 <- rand.ordered[8]
rand9 <- rand.ordered[9]
rand10 <- rand.ordered[10]
rand11 <- rand.ordered[11]
rand12 <- rand.ordered[12]

diff1 <- rand2 - rand1
diff2 <- rand3 - rand2
diff3 <- rand4 - rand3
diff4 <- rand5 - rand4
diff5 <- rand6 - rand5
diff6 <- rand7 - rand6
diff7 <- rand8 - rand7
diff8 <- rand9 - rand8
diff9 <- rand10 - rand9
diff10 <- rand11 - rand10
diff11 <- rand12 - rand11

start1 <- rand1
end1 <- start1 + length(new.do1) - 1

start2 <- rand2
end2 <- start2 + length(new.do2) - 1

start3 <- rand3
end3 <- start3 + length(new.do3) - 1

start4 <- rand4
end4 <- start4 + length(new.do4) - 1

start5 <- rand5
end5 <- start5 + length(new.do5) - 1

start6 <- rand6
end6 <- start6 + length(new.do6) - 1

start7 <- rand7
end7 <- start7 + length(new.do7) - 1

start8 <- rand8
end8 <- start8 + length(new.do8) - 1

start9 <- rand9
end9 <- start9 + length(new.do9) - 1

start10 <- rand10
end10 <- start10 + length(new.do10) - 1

start11 <- rand11
end11 <- start11 + length(new.do11) - 1

start12 <- rand12
end12 <- start12 + length(new.do12) - 1
```

```
d1 <- c(rep(-100, (rand1 - 1)), new.do1, rep(-100, (2000 - (length(new.do1) +
rand1 - 1))))
d2 <- c(rep(-100, (rand2 - 1)), new.do2, rep(-100, (2000 - (length(new.do2) +
rand2 - 1))))
d3 <- c(rep(-100, (rand3 - 1)), new.do3, rep(-100, (2000 - (length(new.do3) +
rand3 - 1))))
d4 <- c(rep(-100, (rand4 - 1)), new.do4, rep(-100, (2000 - (length(new.do4) +
 rand4 - 1))))
d5 <- c(rep(-100, (rand5 - 1)), new.do5, rep(-100, (2000 - (length(new.do5) +
rand5 - 1))))
d6 <- c(rep(-100, (rand6 - 1)), new.do6, rep(-100, (2000 - (length(new.do6) +
rand6 - 1))))
d7 <- c(rep(-100, (rand7 - 1)), new.do7, rep(-100, (2000 - (length(new.do7) +
rand7 - 1))))
d8 <- c(rep(-100, (rand8 - 1)), new.do8, rep(-100, (2000 - (length(new.do8) +
rand8 - 1))))
d9 <- c(rep(-100, (rand9 - 1)), new.do9, rep(-100, (2000 - (length(new.do9) +
rand9 - 1))))
d10 <- c(rep(-100, (rand10 - 1)), new.do10, rep(-100, (2000 - (length(new.do10)
+ rand10 - 1))))
d11 <- c(rep(-100, (rand11 - 1)), new.do11, rep(-100, (2000 - (length(new.do11)
+ rand11 - 1))))
d12 <- c(rep(-100, (rand12 - 1)), new.do12, rep(-100, (2000 - (length(new.do12)
+ rand12 - 1))))

d1.a <- d1[1:1000]
d1.b <- d1[1001:2000]
d2.a <- d2[1:1000]
d2.b <- d2[1001:2000]
d3.a <- d3[1:1000]
d3.b <- d3[1001:2000]
d4.a <- d4[1:1000]
d4.b <- d4[1001:2000]
d5.a <- d5[1:1000]
d5.b <- d5[1001:2000]
d6.a <- d6[1:1000]
d6.b <- d6[1001:2000]
d7.a <- d7[1:1000]
d7.b <- d7[1001:2000]
d8.a <- d8[1:1000]
d8.b <- d8[1001:2000]
d9.a <- d9[1:1000]
d9.b <- d9[1001:2000]
d10.a <- d10[1:1000]
d10.b <- d10[1001:2000]
d11.a <- d11[1:1000]
d11.b <- d11[1001:2000]
d12.a <- d12[1:1000]
d12.b <- d12[1001:2000]


test1 <- pmax(d1.a, d1.b, d2.a, d2.b, d3.a, d3.b, d4.a, d4.b, d5.a, d5.b, d6.a,
d6.b, d7.a, d7.b, d8.a, d8.b, d9.a, d9.b, d10.a, d10.b, d11.a, d11.b, d12.a,
```

```
d12.b)

test1 <- ifelse(test1 == -100, NA, test1)

######################## Filler Points ########################

filler.rand <- sample(1:length(filler), (donum + 1), replace = TRUE)
filler.remain <- filler.rand[1] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[1]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.1 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[2] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[2]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.2 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[3] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[3]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.3 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[4] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}
```

```
filler.per1 <- filler[filler.rand[4]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.4 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[5] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[5]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.5 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[6] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[6]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.6 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[7] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[7]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.7 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[8] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[8]:length(filler)]
```

```
filler.per2 <- filler[filler.obs]
filler.per.8 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[9] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[9]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.9 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[10] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[10]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.10 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[11] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[11]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.11 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[12] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[12]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.12 <- c(filler.per1, filler.per2)
```

```
filler.remain <- filler.rand[13] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[13]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.13 <- c(filler.per1, filler.per2)


filler.ss <- getstartstop(test1)
filler.na <- cbind(rep(NA, 20), rep(NA, 20))
filler.ss1 <- rbind(filler.ss, filler.na)
filler.ss2 <- filler.ss1[1:(donum + 1), ]

fstart1 <- filler.ss2[1, 1]
fend1 <- filler.ss2[1, 2]
flen1 <- fend1 - fstart1 + 1

flen1 <- ifelse(is.na(flen1) == TRUE, 0, flen1)

fstart2 <- filler.ss2[2, 1]
fend2 <- filler.ss2[2, 2]
flen2 <- fend2 - fstart2 + 1

flen2 <- ifelse(is.na(flen2) == TRUE, 0, flen2)

fstart3 <- filler.ss2[3, 1]
fend3 <- filler.ss2[3, 2]
flen3 <- fend3 - fstart3 + 1

flen3 <- ifelse(is.na(flen3) == TRUE, 0, flen3)

fstart4 <- filler.ss2[4, 1]
fend4 <- filler.ss2[4, 2]
flen4 <- fend4 - fstart4 + 1

flen4 <- ifelse(is.na(flen4) == TRUE, 0, flen4)

fstart5 <- filler.ss2[5, 1]
fend5 <- filler.ss2[5, 2]
flen5 <- fend5 - fstart5 + 1

flen5 <- ifelse(is.na(flen5) == TRUE, 0, flen5)

fstart6 <- filler.ss2[6, 1]
fend6 <- filler.ss2[6, 2]
flen6 <- fend6 - fstart6 + 1
```

```
flen6 <- ifelse(is.na(flen6) == TRUE, 0, flen6)

fstart7 <- filler.ss2[7, 1]
fend7 <- filler.ss2[7, 2]
flen7 <- fend7 - fstart7 + 1

flen7 <- ifelse(is.na(flen7) == TRUE, 0, flen7)

fstart8 <- filler.ss2[8, 1]
fend8 <- filler.ss2[8, 2]
flen8 <- fend8 - fstart8 + 1

flen8 <- ifelse(is.na(flen8) == TRUE, 0, flen8)

fstart9 <- filler.ss2[9, 1]
fend9 <- filler.ss2[9, 2]
flen9 <- fend9 - fstart9 + 1

flen9 <- ifelse(is.na(flen9) == TRUE, 0, flen9)

fstart10 <- filler.ss2[10, 1]
fend10 <- filler.ss2[10, 2]
flen10 <- fend10 - fstart10 + 1

flen10 <- ifelse(is.na(flen10) == TRUE, 0, flen10)

fstart11 <- filler.ss2[11, 1]
fend11 <- filler.ss2[11, 2]
flen11 <- fend11 - fstart11 + 1

flen11 <- ifelse(is.na(flen11) == TRUE, 0, flen11)

fstart12 <- filler.ss2[12, 1]
fend12 <- filler.ss2[12, 2]
flen12 <- fend12 - fstart12 + 1

flen12 <- ifelse(is.na(flen12) == TRUE, 0, flen12)

fstart13 <- filler.ss2[13, 1]
fend13 <- filler.ss2[13, 2]
flen13 <- fend13 - fstart13 + 1

flen13 <- ifelse(is.na(flen13) == TRUE, 0, flen13)

rand.filler <- sample(1:(donum + 1), (donum + 1), replace = FALSE)

frand.ordered <- rand.filler[order(rand.filler)]

flrand1 <- rep(frand.ordered[1], flen1)
flrand2 <- rep(frand.ordered[2], flen2)
flrand3 <- rep(frand.ordered[3], flen3)
flrand4 <- rep(frand.ordered[4], flen4)
flrand5 <- rep(frand.ordered[5], flen5)
flrand6 <- rep(frand.ordered[6], flen6)
```

```
flrand7 <- rep(frand.ordered[7], flen7)
flrand8 <- rep(frand.ordered[8], flen8)
flrand9 <- rep(frand.ordered[9], flen9)
flrand10 <- rep(frand.ordered[10], flen10)
flrand11 <- rep(frand.ordered[11], flen11)
flrand12 <- rep(frand.ordered[12], flen12)
flrand13 <- rep(frand.ordered[13], flen13)

flrand1 <- as.matrix(flrand1)
flrand2 <- as.matrix(flrand2)
flrand3 <- as.matrix(flrand3)
flrand4 <- as.matrix(flrand4)
flrand5 <- as.matrix(flrand5)
flrand6 <- as.matrix(flrand6)
flrand7 <- as.matrix(flrand7)
flrand8 <- as.matrix(flrand8)
flrand9 <- as.matrix(flrand9)
flrand10 <- as.matrix(flrand10)
flrand11 <- as.matrix(flrand11)
flrand12 <- as.matrix(flrand12)
flrand13 <- as.matrix(flrand13)

flrand <- rbind(flrand1, flrand2, flrand3, flrand4, flrand5, flrand6, flrand7,
flrand8, flrand9, flrand10, flrand11, flrand12, flrand13)

obs.frand1 <- which(flrand == rand.filler[1])
obs.frand2 <- which(flrand == rand.filler[2])
obs.frand3 <- which(flrand == rand.filler[3])
obs.frand4 <- which(flrand == rand.filler[4])
obs.frand5 <- which(flrand == rand.filler[5])
obs.frand6 <- which(flrand == rand.filler[6])
obs.frand7 <- which(flrand == rand.filler[7])
obs.frand8 <- which(flrand == rand.filler[8])
obs.frand9 <- which(flrand == rand.filler[9])
obs.frand10 <- which(flrand == rand.filler[10])
obs.frand11 <- which(flrand == rand.filler[11])
obs.frand12 <- which(flrand == rand.filler[12])
obs.frand13 <- which(flrand == rand.filler[13])

if(length(obs.frand1) == 0){
obs.frand1 <- 0
}else
{
obs.frand1 <- min(obs.frand1):max(obs.frand1)
}

if(length(obs.frand2) == 0){
obs.frand2 <- 0
}else
{
obs.frand2 <- min(obs.frand2):max(obs.frand2)
}

if(length(obs.frand3) == 0){
```

```
obs.frand3 <- 0
}else
{
obs.frand3 <- min(obs.frand3):max(obs.frand3)
}

if(length(obs.frand4) == 0){
obs.frand4 <- 0
}else
{
obs.frand4 <- min(obs.frand4):max(obs.frand4)
}

if(length(obs.frand5) == 0){
obs.frand5 <- 0
}else
{
obs.frand5 <- min(obs.frand5):max(obs.frand5)
}

if(length(obs.frand6) == 0){
obs.frand6 <- 0
}else
{
obs.frand6 <- min(obs.frand6):max(obs.frand6)
}

if(length(obs.frand7) == 0){
obs.frand7 <- 0
}else
{
obs.frand7 <- min(obs.frand7):max(obs.frand7)
}

if(length(obs.frand8) == 0){
obs.frand8 <- 0
}else
{
obs.frand8 <- min(obs.frand8):max(obs.frand8)
}

if(length(obs.frand9) == 0){
obs.frand9 <- 0
}else
{
obs.frand9 <- min(obs.frand9):max(obs.frand9)
}

if(length(obs.frand10) == 0){
obs.frand10 <- 0
}else
{
obs.frand10 <- min(obs.frand10):max(obs.frand10)
}
```

```
if(length(obs.frand11) == 0){
obs.frand11 <- 0
}else
{
obs.frand11 <- min(obs.frand11):max(obs.frand11)
}

if(length(obs.frand12) == 0){
obs.frand12 <- 0
}else
{
obs.frand12 <- min(obs.frand12):max(obs.frand12)
}

if(length(obs.frand13) == 0){
obs.frand13 <- 0
}else
{
obs.frand13 <- min(obs.frand13):max(obs.frand13)
}


frand1 <- flrand[obs.frand1, ]
frand2 <- flrand[obs.frand2, ]
frand3 <- flrand[obs.frand3, ]
frand4 <- flrand[obs.frand4, ]
frand5 <- flrand[obs.frand5, ]
frand6 <- flrand[obs.frand6, ]
frand7 <- flrand[obs.frand7, ]
frand8 <- flrand[obs.frand8, ]
frand9 <- flrand[obs.frand9, ]
frand10 <- flrand[obs.frand10, ]
frand11 <- flrand[obs.frand11, ]
frand12 <- flrand[obs.frand12, ]
frand13 <- flrand[obs.frand13, ]

frand1 <- as.matrix(frand1)
frand2 <- as.matrix(frand2)
frand3 <- as.matrix(frand3)
frand4 <- as.matrix(frand4)
frand5 <- as.matrix(frand5)
frand6 <- as.matrix(frand6)
frand7 <- as.matrix(frand7)
frand8 <- as.matrix(frand8)
frand9 <- as.matrix(frand9)
frand10 <- as.matrix(frand10)
frand11 <- as.matrix(frand11)
frand12 <- as.matrix(frand12)
frand13 <- as.matrix(frand13)

frand <- rbind(frand1, frand2, frand3, frand4, frand5, frand6, frand7,frand8,
frand9, frand10, frand11, frand12, frand13)
```

```
c.frand <- cbind(frand, filler.per.1, filler.per.2, filler.per.3, filler.per.4,
filler.per.5, filler.per.6, filler.per.7, filler.per.8, filler.per.9,
filler.per.10, filler.per.11, filler.per.12, filler.per.13)

fordered <- c.frand[order(c.frand[, 1]), ]

obs.f1 <- which(fordered[ , 1] == frand.ordered[1])
obs.f2 <- which(fordered[ , 1] == frand.ordered[2])
obs.f3 <- which(fordered[ , 1] == frand.ordered[3])
obs.f4 <- which(fordered[ , 1] == frand.ordered[4])
obs.f5 <- which(fordered[ , 1] == frand.ordered[5])
obs.f6 <- which(fordered[ , 1] == frand.ordered[6])
obs.f7 <- which(fordered[ , 1] == frand.ordered[7])
obs.f8 <- which(fordered[ , 1] == frand.ordered[8])
obs.f9 <- which(fordered[ , 1] == frand.ordered[9])
obs.f10 <- which(fordered[ , 1] == frand.ordered[10])
obs.f11 <- which(fordered[ , 1] == frand.ordered[11])
obs.f12 <- which(fordered[ , 1] == frand.ordered[12])
obs.f13 <- which(fordered[ , 1] == frand.ordered[13])

if(length(obs.f1) == 0){
obs.f1 <- 0
}else
{
obs.f1 <- min(obs.f1):max(obs.f1)
}

if(length(obs.f2) == 0){
obs.f2 <- 0
}else
{
obs.f2 <- min(obs.f2):max(obs.f2)
}

if(length(obs.f3) == 0){
obs.f3 <- 0
}else
{
obs.f3 <- min(obs.f3):max(obs.f3)
}

if(length(obs.f4) == 0){
obs.f4 <- 0
}else
{
obs.f4 <- min(obs.f4):max(obs.f4)
}

if(length(obs.f5) == 0){
obs.f5 <- 0
}else
{
obs.f5 <- min(obs.f5):max(obs.f5)
}
```

```
if(length(obs.f6) == 0){
obs.f6 <- 0
}else
{
obs.f6 <- min(obs.f6):max(obs.f6)
}

if(length(obs.f7) == 0){
obs.f7 <- 0
}else
{
obs.f7 <- min(obs.f7):max(obs.f7)
}

if(length(obs.f8) == 0){
obs.f8 <- 0
}else
{
obs.f8 <- min(obs.f8):max(obs.f8)
}

if(length(obs.f9) == 0){
obs.f9 <- 0
}else
{
obs.f9 <- min(obs.f9):max(obs.f9)
}

if(length(obs.f10) == 0){
obs.f10 <- 0
}else
{
obs.f10 <- min(obs.f10):max(obs.f10)
}

if(length(obs.f11) == 0){
obs.f11 <- 0
}else
{
obs.f11 <- min(obs.f11):max(obs.f11)
}

if(length(obs.f12) == 0){
obs.f12 <- 0
}else
{
obs.f12 <- min(obs.f12):max(obs.f12)
}

if(length(obs.f13) == 0){
obs.frand13 <- 0
}else
{
```

```
obs.f13 <- min(obs.f13):max(obs.f13)
}


new.f1 <- fordered[obs.f1, 2]
new.f2 <- fordered[obs.f2, 3]
new.f3 <- fordered[obs.f3, 4]
new.f4 <- fordered[obs.f4, 5]
new.f5 <- fordered[obs.f5, 6]
new.f6 <- fordered[obs.f6, 7]
new.f7 <- fordered[obs.f7, 8]
new.f8 <- fordered[obs.f8, 9]
new.f9 <- fordered[obs.f9, 10]
new.f10 <- fordered[obs.f10, 11]
new.f11 <- fordered[obs.f11, 12]
new.f12 <- fordered[obs.f12, 13]
new.f13 <- fordered[obs.f13, 14]

if(length(new.f1) == 0){
f1.obs <- 0
}else
{
f1.obs <- fstart1:fend1
}

if(length(new.f2) == 0){
f2.obs <- 0
}else
{
f2.obs <- fstart2:fend2
}

if(length(new.f3) == 0){
f3.obs <- 0
}else
{
f3.obs <- fstart3:fend3
}

if(length(new.f4) == 0){
f4.obs <- 0
}else
{
f4.obs <- fstart4:fend4
}

if(length(new.f5) == 0){
f5.obs <- 0
}else
{
f5.obs <- fstart5:fend5
}

if(length(new.f6) == 0){
```

```
f6.obs <- 0
}else
{
f6.obs <- fstart6:fend6
}

if(length(new.f7) == 0){
f7.obs <- 0
}else
{
f7.obs <- fstart7:fend7
}

if(length(new.f8) == 0){
f8.obs <- 0
}else
{
f8.obs <- fstart8:fend8
}

if(length(new.f9) == 0){
f9.obs <- 0
}else
{
f9.obs <- fstart9:fend9
}

if(length(new.f10) == 0){
f10.obs <- 0
}else
{
f10.obs <- fstart10:fend10
}

if(length(new.f11) == 0){
f11.obs <- 0
}else
{
f11.obs <- fstart11:fend11
}

if(length(new.f12) == 0){
f12.obs <- 0
}else
{
f12.obs <- fstart12:fend12
}

if(length(new.f13) == 0){
f13.obs <- 0
}else
{
f13.obs <- fstart13:fend13
}
```

```
test1[f1.obs] <- new.f1
test1[f2.obs] <- new.f2
test1[f3.obs] <- new.f3
test1[f4.obs] <- new.f4
test1[f5.obs] <- new.f5
test1[f6.obs] <- new.f6
test1[f7.obs] <- new.f7
test1[f8.obs] <- new.f8
test1[f9.obs] <- new.f9
test1[f10.obs] <- new.f10
test1[f11.obs] <- new.f11
test1[f12.obs] <- new.f12
test1[f13.obs] <- new.f13

# Scaling test1
test1 <- test1/sqrt(var(test1))

################## End of D-O Event Scattering ##################

#################################################################
########### Periodogram for Block Bootstrap of GISP2 ############
#################################################################

sim.fs <- cbind(sim.p.gram$freq, sim.p.gram$spec)
sim.max.loc <- which.max(sim.fs[ , 2])


sim.gmax <- sim.fs[sim.max.loc, ]
all.sim.gmax <- rbind(all.sim.gmax, sim.gmax)

sim.fshort.loc1 <- which(sim.fs[ , 1] <= 0.1)
sim.fshort.loc2 <- max(sim.fshort.loc1)
sim.fs.short <- sim.fs[1:sim.fshort.loc2, ]

sim.spec <- as.matrix(sim.fs.short[ , 2])
sim.data1 <- rbind(0, sim.spec, 0)
sim.data1 <- as.numeric(sim.data1)
sim.lmax.spec <- rep(NA, length(sim.data1))
for(i in 1:length(sim.data1)){
if(sim.data1[i] >= sim.data1[i+1] && sim.data1[i] >= sim.data1[i-1])
{sim.lmax.spec[i] = sim.data1[i]}
}

sim.loc.na <- is.na(sim.lmax.spec)
sim.non.na <- which(sim.loc.na == "FALSE")
sim.loc.non.na <- sim.non.na - 1

sim.lmax <- sim.fs.short[sim.loc.non.na, ]
all.sim.lmax <- rbind(all.sim.lmax, cbind(j, sim.lmax))

ordered <- all.sim.lmax[order(all.sim.lmax[ , 3], decreasing = TRUE), ]

####### End of Periodogram for Block Bootstrap of GISP2 ########
```

```
#####################################################################
#################### Statistic of Interest ##########################
#####################################################################

order.freq <- sim.lmax[order(sim.lmax[, 1]), ]
DO.data <- matrix(order.freq, byrow = FALSE, ncol = 2)
DO.data <- cbind(j, DO.data)
DO.stat.full <- rbind(DO.stat.full, DO.data)

################## End of Statistic of Interest ################

}

####################### End of Simulation ######################

write.table(DO.stat.full, file = "/Users/andrea/school/research/URGES/Thesis
Project/Aspect 1/Permutation Test Code/Final Product/Simulations2/Results/
Duration/Null Distribution/DOstat full for duration 0.25.txt", row.names =
FALSE, col.names = c("simnum", "freq", "spec"))

#####################################################################
###### Future Analysis without having to run simulation #######
#####################################################################

dostat <- read.table("/Users/andrea/school/research/URGES/Thesis Project/Aspect
1/Permutation Test Code/Final Product/Simulations2/Results/Duration/Null
Distribution/DOstat full for duration 0.25.txt", header = TRUE)

# Cyclicity less than 3000 (> 0.00909)

DO.stat <- NULL
numsim <- 10000
for(i in 1:numsim){

obs.num <- which(dostat$simnum == i)

chunk <- dostat[min(obs.num):max(obs.num), ]

chunk <- chunk[, -1]

stat.wind <- which(chunk$freq > 0.00909)
dostat.data <- chunk[min(stat.wind):max(stat.wind), ]
dostat.data <- as.matrix(dostat.data)

chunk.max <- which.max(dostat.data[, 2])

part.max <- dostat.data[chunk.max, ]
DO.stat <- rbind(DO.stat, part.max)

}

write.table(DO.stat, file = "/Users/andrea/school/research/URGES/Thesis Project/
Aspect 1/Permutation Test Code/Final Product/Simulations2/Results/Duration/Null
```

```
Distribution/DOstat 3000 for duration 0.25.txt", row.names = FALSE, col.names =
c("freq", "spec"))

##############################################################################
##############################################################################
##############################################################################
# Skipped Manifestations of D-O events
##############################################################################
##############################################################################
##############################################################################
# To be used after creating pseudo-series, which are called full.per.t
  in this code

full.per.t <- matrix(scan("/Users/andrea/Desktop/full periodograms for
duration 0.25.txt", sep = " "), nrow = 500, ncol = 10000, byrow = TRUE)

full.per <- t(full.per.t)

#################################################################

my.freq <- function(ip.dist, cyc){

freq <- ip.dist/cyc
freq

}

my.cyc <- function(ip.dist, freq.inc){

cyc <- (1/freq.inc)*ip.dist
cyc

}

ip.dist <- 27.27
d <- 0.15
x <- 0:100

freq.inc <- 0.109 - x * 0.001
freq.inc <- as.numeric(as.character(freq.inc))
freq.inc.1000 <- freq.inc * 1000

freq2cyc <- my.cyc(ip.dist, freq.inc)

l <- 2 * freq2cyc - d * 2 * freq2cyc
u <- 2 * freq2cyc + d * 2 * freq2cyc

cyc2freq.u <- my.freq(ip.dist, u) * 1000
cyc2freq.l <- my.freq(ip.dist, l) * 1000

numsim <- 10000
full.sd <- NULL
full.max.sd <- NULL
```

```
for(i in 1:numsim){

sum.duple <- NULL
for(j in 1:101){

p1 <- full.per[i, freq.inc.1000[j]]
col.obs <- ceiling(cyc2freq.u[j]):floor(cyc2freq.l[j])

part.per <- full.per[i, col.obs]
max.obs <- which.max(part.per)
p2 <- part.per[max.obs]

sum.p1p2 <- sum(p1, p2)
sum.duple1 <- cbind(freq.inc[j], (col.obs[max.obs]/1000), sum.p1p2)
sum.duple <- rbind(sum.duple, sum.duple1)

}

max.sd <- max(sum.duple[, 3])
full.sd <- rbind(full.sd, sum.duple)
full.max.sd <- rbind(full.max.sd, max.sd)

}

write.table(full.max.sd, file = "/Users/andrea/school/research/URGES/Thesis
Project/Aspect 1/Permutation Test Code/Final Product/Simulations2/Results/
Duration/Ideal D-O event/max sd for duration 0.25.txt", row.names = FALSE,
col.name = "Max")

write.table(full.sd, file = "/Users/andrea/school/research/URGES/Thesis Project/
Aspect 1/Permutation Test Code/Final Product/Simulations2/Results/Duration/Ideal
D-O event/sd for duration 0.25.txt", row.names = FALSE, col.names = TRUE)
```

C.2.5 Averaged p-values

```
# Sample code of finding the averaged p-values

ave.pval <- function(do.event, do.ts){
p.val <- NULL
for(i in 1:numsim){

num.obs <- which(do.event$Max >= do.ts[i, ])
num <- length(num.obs) + 1

p.val.part <- num/(numsim + 1)

p.val <- rbind(p.val, p.val.part)

}
mean(p.val)
}

numsim <- 10000
```

```
# Pseudo GISP2
# cycles < 2,500 years
g20 <- read.table("./scaled DOstat 2500 for 47 to 20 kyr BP.txt", header = TRUE)

# cycles < 3,000 years
g20 <- read.table("./scaled DOstat 3000 for 47 to 20 kyr BP.txt", header = TRUE)

# cycles < 5,000 years
g20 <- read.table("./scaled DOstat 5000 for 47 to 20 kyr BP.txt", header = TRUE)

# duple
g20 <- read.table("./ll max sd for 47 to 20 kyr BP.txt", header = TRUE)

# Test statistic datasets
do.ts2500.20 <- read.table("./scaled DOstat 2500 for duration 1.00.txt", header
= TRUE)
do.ts3000.20 <- read.table("./scaled DOstat 3000 for duration 1.00.txt", header
= TRUE)
do.ts5000.20 <- read.table("./scaled DOstat 5000 for duration 1.00.txt", header
= TRUE)

do.ts.20 <- read.table("./ll max sd for duration 1.00.txt", header = TRUE)

ave.pval(g20, do.ts2500.20)
ave.pval(g20, do.ts3000.20)
ave.pval(g20, do.ts5000.20)
ave.pval(g20, do.ts.20)
```

C.3 Correlation Between the Greenland and Antarctica Records

C.3.1 Null Distribution and Test Statistic

```
gisp2 <- read.table(".../GISP2 47-20 kyr detrend interp.txt", header = TRUE)

edml <- read.table(".../EDML 47-20 kyr detrend interp.txt", header = TRUE)

#####################################################################
#################### Corr for GISP2 and EDML ########################
#####################################################################

all.cor <- ccf(gisp2$res, edml$res, type = "correlation", plot =FALSE, lag.max
= 92)
orig.cor <- cbind(all.cor$lag, all.cor$acf)

################## End of Corr for GISP2 and EDML ###############

getstartstop <- function(j1)
{
  j1b <- c(j1[2:(length(j1))],-999999)
  j1f <- c(-999999,j1[1:(length(j1)-1)])
  starts <- is.na(j1) & !is.na(j1f)
  stops <- is.na(j1) & !is.na(j1b)
  #cbind(starts,stops)  # use this if you want an n by 2 matrix of T's and F's
```

```
  ## or ##
  cbind( (1:length(j1))[starts] , (1:length(j1))[stops] )
      # use this if you want a k by 2 matrix of start and stop locations
}

#####################################################################
#####################################################################
#####################################################################
########################## Simulation ###############################
#####################################################################
#####################################################################
#####################################################################

numsim <- 10000
numrows <- length(all.cor$acf)
sim.cor <- matrix(NA, numrows, numsim)
#sim.cor <- NULL
max.lag <- NULL
max.abs <- NULL
cor.0 <- NULL

for(j in 1:numsim){

#####################################################################
################### D-O Event Scattering#############################
#####################################################################

do2 <- gisp2[119:129, 2]
do3 <- gisp2[272:292, 2]
do4 <- gisp2[316:337, 2]
do5 <- gisp2[423:455, 2]
do6 <- gisp2[475:516, 2]
do7 <- gisp2[521:563, 2]
do8 <- gisp2[581:677, 2]
do9 <- gisp2[731:744, 2]
do10 <- gisp2[755:777, 2]
do11 <- gisp2[786:829, 2]
do12 <- gisp2[834:933, 2]
do13 <- gisp2[966:998, 2]

filler <- gisp2[ , 2]
filler[119:129] <- NA
filler[272:292] <- NA
filler[316:337] <- NA
filler[423:455] <- NA
filler[475:516] <- NA
filler[521:563] <- NA
filler[581:677] <- NA
filler[731:744] <- NA
filler[755:777] <- NA
filler[786:829] <- NA
filler[834:933] <- NA
filler[966:998] <- NA
```

```
filler <- na.omit(filler)

rand.do <- sample(2:13, 12, replace = FALSE)

do2.rand <- cbind(do2, 2)
do3.rand <- cbind(do3, 3)
do4.rand <- cbind(do4, 4)
do5.rand <- cbind(do5, 5)
do6.rand <- cbind(do6, 6)
do7.rand <- cbind(do7, 7)
do8.rand <- cbind(do8, 8)
do9.rand <- cbind(do9, 9)
do10.rand <- cbind(do10, 10)
do11.rand <- cbind(do11, 11)
do12.rand <- cbind(do12, 12)
do13.rand <- cbind(do13, 13)

do.rand <- rbind(do2.rand, do3.rand, do4.rand, do5.rand, do6.rand, do7.rand,
do8.rand, do9.rand, do10.rand, do11.rand, do12.rand, do13.rand)

obs.do1.rand <- which(do.rand[ , 2] == rand.do[1])
obs.do2.rand <- which(do.rand[ , 2] == rand.do[2])
obs.do3.rand <- which(do.rand[ , 2] == rand.do[3])
obs.do4.rand <- which(do.rand[ , 2] == rand.do[4])
obs.do5.rand <- which(do.rand[ , 2] == rand.do[5])
obs.do6.rand <- which(do.rand[ , 2] == rand.do[6])
obs.do7.rand <- which(do.rand[ , 2] == rand.do[7])
obs.do8.rand <- which(do.rand[ , 2] == rand.do[8])
obs.do9.rand <- which(do.rand[ , 2] == rand.do[9])
obs.do10.rand <- which(do.rand[ , 2] == rand.do[10])
obs.do11.rand <- which(do.rand[ , 2] == rand.do[11])
obs.do12.rand <- which(do.rand[ , 2] == rand.do[12])

new.do1.rand <- do.rand[min(obs.do1.rand):max(obs.do1.rand), 1]
new.do2.rand <- do.rand[min(obs.do2.rand):max(obs.do2.rand), 1]
new.do3.rand <- do.rand[min(obs.do3.rand):max(obs.do3.rand), 1]
new.do4.rand <- do.rand[min(obs.do4.rand):max(obs.do4.rand), 1]
new.do5.rand <- do.rand[min(obs.do5.rand):max(obs.do5.rand), 1]
new.do6.rand <- do.rand[min(obs.do6.rand):max(obs.do6.rand), 1]
new.do7.rand <- do.rand[min(obs.do7.rand):max(obs.do7.rand), 1]
new.do8.rand <- do.rand[min(obs.do8.rand):max(obs.do8.rand), 1]
new.do9.rand <- do.rand[min(obs.do9.rand):max(obs.do9.rand), 1]
new.do10.rand <- do.rand[min(obs.do10.rand):max(obs.do10.rand), 1]
new.do11.rand <- do.rand[min(obs.do11.rand):max(obs.do11.rand), 1]
new.do12.rand <- do.rand[min(obs.do12.rand):max(obs.do12.rand), 1]

donum <- 12

do.obs <- sum(length(new.do1.rand), length(new.do2.rand), length(new.do3.rand),
length(new.do4.rand), length(new.do5.rand), length(new.do6.rand),
length(new.do7.rand), length(new.do8.rand), length(new.do9.rand),
length(new.do10.rand), length(new.do11.rand), length(new.do12.rand))

filler.num <- length(gisp2$age)
```

```
rand.nums <- sample(1:filler.num, donum, replace = FALSE)

do1.1 <- cbind(rand.nums[1], new.do1.rand)
do2.1 <- cbind(rand.nums[2], new.do2.rand)
do3.1 <- cbind(rand.nums[3], new.do3.rand)
do4.1 <- cbind(rand.nums[4], new.do4.rand)
do5.1 <- cbind(rand.nums[5], new.do5.rand)
do6.1 <- cbind(rand.nums[6], new.do6.rand)
do7.1 <- cbind(rand.nums[7], new.do7.rand)
do8.1 <- cbind(rand.nums[8], new.do8.rand)
do9.1 <- cbind(rand.nums[9], new.do9.rand)
do10.1 <- cbind(rand.nums[10], new.do10.rand)
do11.1 <- cbind(rand.nums[11], new.do11.rand)
do12.1 <- cbind(rand.nums[12], new.do12.rand)


doevents <- rbind(do1.1, do2.1, do3.1, do4.1, do5.1, do6.1, do7.1, do8.1, do9.1,
do10.1, do11.1, do12.1)

ordered <- doevents[order(doevents[, 1]), ]
rand.ordered <- rand.nums[order(rand.nums)]

obs.do1 <- which(ordered[ , 1] == rand.ordered[1])
obs.do2 <- which(ordered[ , 1] == rand.ordered[2])
obs.do3 <- which(ordered[ , 1] == rand.ordered[3])
obs.do4 <- which(ordered[ , 1] == rand.ordered[4])
obs.do5 <- which(ordered[ , 1] == rand.ordered[5])
obs.do6 <- which(ordered[ , 1] == rand.ordered[6])
obs.do7 <- which(ordered[ , 1] == rand.ordered[7])
obs.do8 <- which(ordered[ , 1] == rand.ordered[8])
obs.do9 <- which(ordered[ , 1] == rand.ordered[9])
obs.do10 <- which(ordered[ , 1] == rand.ordered[10])
obs.do11 <- which(ordered[ , 1] == rand.ordered[11])
obs.do12 <- which(ordered[ , 1] == rand.ordered[12])

new.do1 <- ordered[min(obs.do1):max(obs.do1), 2]
new.do2 <- ordered[min(obs.do2):max(obs.do2), 2]
new.do3 <- ordered[min(obs.do3):max(obs.do3), 2]
new.do4 <- ordered[min(obs.do4):max(obs.do4), 2]
new.do5 <- ordered[min(obs.do5):max(obs.do5), 2]
new.do6 <- ordered[min(obs.do6):max(obs.do6), 2]
new.do7 <- ordered[min(obs.do7):max(obs.do7), 2]
new.do8 <- ordered[min(obs.do8):max(obs.do8), 2]
new.do9 <- ordered[min(obs.do9):max(obs.do9), 2]
new.do10 <- ordered[min(obs.do10):max(obs.do10), 2]
new.do11 <- ordered[min(obs.do11):max(obs.do11), 2]
new.do12 <- ordered[min(obs.do12):max(obs.do12), 2]

rand1 <- rand.ordered[1]
rand2 <- rand.ordered[2]
rand3 <- rand.ordered[3]
rand4 <- rand.ordered[4]
rand5 <- rand.ordered[5]
```

```
rand6 <- rand.ordered[6]
rand7 <- rand.ordered[7]
rand8 <- rand.ordered[8]
rand9 <- rand.ordered[9]
rand10 <- rand.ordered[10]
rand11 <- rand.ordered[11]
rand12 <- rand.ordered[12]

diff1 <- rand2 - rand1
diff2 <- rand3 - rand2
diff3 <- rand4 - rand3
diff4 <- rand5 - rand4
diff5 <- rand6 - rand5
diff6 <- rand7 - rand6
diff7 <- rand8 - rand7
diff8 <- rand9 - rand8
diff9 <- rand10 - rand9
diff10 <- rand11 - rand10
diff11 <- rand12 - rand11

start1 <- rand1
end1 <- start1 + length(new.do1) - 1

start2 <- rand2
end2 <- start2 + length(new.do2) - 1

start3 <- rand3
end3 <- start3 + length(new.do3) - 1

start4 <- rand4
end4 <- start4 + length(new.do4) - 1

start5 <- rand5
end5 <- start5 + length(new.do5) - 1

start6 <- rand6
end6 <- start6 + length(new.do6) - 1

start7 <- rand7
end7 <- start7 + length(new.do7) - 1

start8 <- rand8
end8 <- start8 + length(new.do8) - 1

start9 <- rand9
end9 <- start9 + length(new.do9) - 1

start10 <- rand10
end10 <- start10 + length(new.do10) - 1

start11 <- rand11
end11 <- start11 + length(new.do11) - 1

start12 <- rand12
```

```
end12 <- start12 + length(new.do12) - 1


d1 <- c(rep(-100, (rand1 - 1)), new.do1, rep(-100, (2000 - (length(new.do1) +
rand1 - 1))))

d2 <- c(rep(-100, (rand2 - 1)), new.do2, rep(-100, (2000 - (length(new.do2) +
rand2 - 1))))

d3 <- c(rep(-100, (rand3 - 1)), new.do3, rep(-100, (2000 - (length(new.do3) +
rand3 - 1))))

d4 <- c(rep(-100, (rand4 - 1)), new.do4, rep(-100, (2000 - (length(new.do4) +
rand4 - 1))))

d5 <- c(rep(-100, (rand5 - 1)), new.do5, rep(-100, (2000 - (length(new.do5) +
rand5 - 1))))

d6 <- c(rep(-100, (rand6 - 1)), new.do6, rep(-100, (2000 - (length(new.do6) +
rand6 - 1))))

d7 <- c(rep(-100, (rand7 - 1)), new.do7, rep(-100, (2000 - (length(new.do7) +
rand7 - 1))))

d8 <- c(rep(-100, (rand8 - 1)), new.do8, rep(-100, (2000 - (length(new.do8) +
rand8 - 1))))

d9 <- c(rep(-100, (rand9 - 1)), new.do9, rep(-100, (2000 - (length(new.do9) +
rand9 - 1))))

d10 <- c(rep(-100, (rand10 - 1)), new.do10, rep(-100, (2000 - (length(new.do10)
+ rand10 - 1))))

d11 <- c(rep(-100, (rand11 - 1)), new.do11, rep(-100, (2000 - (length(new.do11)
+ rand11 - 1))))

d12 <- c(rep(-100, (rand12 - 1)), new.do12, rep(-100, (2000 - (length(new.do12)
+ rand12 - 1))))

d1.a <- d1[1:1000]
d1.b <- d1[1001:2000]
d2.a <- d2[1:1000]
d2.b <- d2[1001:2000]
d3.a <- d3[1:1000]
d3.b <- d3[1001:2000]
d4.a <- d4[1:1000]
d4.b <- d4[1001:2000]
d5.a <- d5[1:1000]
d5.b <- d5[1001:2000]
d6.a <- d6[1:1000]
d6.b <- d6[1001:2000]
d7.a <- d7[1:1000]
d7.b <- d7[1001:2000]
d8.a <- d8[1:1000]
```

```
d8.b <- d8[1001:2000]
d9.a <- d9[1:1000]
d9.b <- d9[1001:2000]
d10.a <- d10[1:1000]
d10.b <- d10[1001:2000]
d11.a <- d11[1:1000]
d11.b <- d11[1001:2000]
d12.a <- d12[1:1000]
d12.b <- d12[1001:2000]

test1 <- pmax(d1.a, d1.b, d2.a, d2.b, d3.a, d3.b, d4.a, d4.b, d5.a, d5.b, d6.a,
d6.b, d7.a, d7.b, d8.a, d8.b, d9.a, d9.b, d10.a, d10.b, d11.a, d11.b, d12.a,
d12.b)

test1 <- ifelse(test1 == -100, NA, test1)

######################### Filler Points ##########################

filler.rand <- sample(1:length(filler), (donum + 1), replace = TRUE)

filler.remain <- filler.rand[1] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[1]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.1 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[2] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[2]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.2 <- c(filler.per1, filler.per2)


filler.remain <- filler.rand[3] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
```

```
}

filler.per1 <- filler[filler.rand[3]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.3 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[4] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[4]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.4 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[5] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[5]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.5 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[6] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[6]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.6 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[7] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}
```

```
filler.per1 <- filler[filler.rand[7]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.7 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[8] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[8]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.8 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[9] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[9]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.9 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[10] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[10]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.10 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[11] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[11]:length(filler)]
filler.per2 <- filler[filler.obs]
```

```
filler.per.11 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[12] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[12]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.12 <- c(filler.per1, filler.per2)

filler.remain <- filler.rand[13] - 1

if(filler.remain == 0){
filler.obs <- 0
} else
{
filler.obs <- 1:filler.remain
}

filler.per1 <- filler[filler.rand[13]:length(filler)]
filler.per2 <- filler[filler.obs]
filler.per.13 <- c(filler.per1, filler.per2)

filler.ss <- getstartstop(test1)
filler.na <- cbind(rep(NA, 20), rep(NA, 20))
filler.ss1 <- rbind(filler.ss, filler.na)
filler.ss2 <- filler.ss1[1:(donum + 1), ]

fstart1 <- filler.ss2[1, 1]
fend1 <- filler.ss2[1, 2]
flen1 <- fend1 - fstart1 + 1

flen1 <- ifelse(is.na(flen1) == TRUE, 0, flen1)

fstart2 <- filler.ss2[2, 1]
fend2 <- filler.ss2[2, 2]
flen2 <- fend2 - fstart2 + 1

flen2 <- ifelse(is.na(flen2) == TRUE, 0, flen2)

fstart3 <- filler.ss2[3, 1]
fend3 <- filler.ss2[3, 2]
flen3 <- fend3 - fstart3 + 1

flen3 <- ifelse(is.na(flen3) == TRUE, 0, flen3)

fstart4 <- filler.ss2[4, 1]
fend4 <- filler.ss2[4, 2]
flen4 <- fend4 - fstart4 + 1
```

```
flen4 <- ifelse(is.na(flen4) == TRUE, 0, flen4)

fstart5 <- filler.ss2[5, 1]
fend5 <- filler.ss2[5, 2]
flen5 <- fend5 - fstart5 + 1

flen5 <- ifelse(is.na(flen5) == TRUE, 0, flen5)

fstart6 <- filler.ss2[6, 1]
fend6 <- filler.ss2[6, 2]
flen6 <- fend6 - fstart6 + 1

flen6 <- ifelse(is.na(flen6) == TRUE, 0, flen6)

fstart7 <- filler.ss2[7, 1]
fend7 <- filler.ss2[7, 2]
flen7 <- fend7 - fstart7 + 1

flen7 <- ifelse(is.na(flen7) == TRUE, 0, flen7)

fstart8 <- filler.ss2[8, 1]
fend8 <- filler.ss2[8, 2]
flen8 <- fend8 - fstart8 + 1

flen8 <- ifelse(is.na(flen8) == TRUE, 0, flen8)

fstart9 <- filler.ss2[9, 1]
fend9 <- filler.ss2[9, 2]
flen9 <- fend9 - fstart9 + 1

flen9 <- ifelse(is.na(flen9) == TRUE, 0, flen9)

fstart10 <- filler.ss2[10, 1]
fend10 <- filler.ss2[10, 2]
flen10 <- fend10 - fstart10 + 1

flen10 <- ifelse(is.na(flen10) == TRUE, 0, flen10)

fstart11 <- filler.ss2[11, 1]
fend11 <- filler.ss2[11, 2]
flen11 <- fend11 - fstart11 + 1

flen11 <- ifelse(is.na(flen11) == TRUE, 0, flen11)

fstart12 <- filler.ss2[12, 1]
fend12 <- filler.ss2[12, 2]
flen12 <- fend12 - fstart12 + 1

flen12 <- ifelse(is.na(flen12) == TRUE, 0, flen12)

fstart13 <- filler.ss2[13, 1]
fend13 <- filler.ss2[13, 2]
flen13 <- fend13 - fstart13 + 1
```

```
flen13 <- ifelse(is.na(flen13) == TRUE, 0, flen13)

rand.filler <- sample(1:(donum + 1), (donum + 1), replace = FALSE)

frand.ordered <- rand.filler[order(rand.filler)]

flrand1 <- rep(frand.ordered[1], flen1)
flrand2 <- rep(frand.ordered[2], flen2)
flrand3 <- rep(frand.ordered[3], flen3)
flrand4 <- rep(frand.ordered[4], flen4)
flrand5 <- rep(frand.ordered[5], flen5)
flrand6 <- rep(frand.ordered[6], flen6)
flrand7 <- rep(frand.ordered[7], flen7)
flrand8 <- rep(frand.ordered[8], flen8)
flrand9 <- rep(frand.ordered[9], flen9)
flrand10 <- rep(frand.ordered[10], flen10)
flrand11 <- rep(frand.ordered[11], flen11)
flrand12 <- rep(frand.ordered[12], flen12)
flrand13 <- rep(frand.ordered[13], flen13)

flrand1 <- as.matrix(flrand1)
flrand2 <- as.matrix(flrand2)
flrand3 <- as.matrix(flrand3)
flrand4 <- as.matrix(flrand4)
flrand5 <- as.matrix(flrand5)
flrand6 <- as.matrix(flrand6)
flrand7 <- as.matrix(flrand7)
flrand8 <- as.matrix(flrand8)
flrand9 <- as.matrix(flrand9)
flrand10 <- as.matrix(flrand10)
flrand11 <- as.matrix(flrand11)
flrand12 <- as.matrix(flrand12)
flrand13 <- as.matrix(flrand13)

flrand <- rbind(flrand1, flrand2, flrand3, flrand4, flrand5, flrand6, flrand7,
flrand8, flrand9, flrand10, flrand11, flrand12, flrand13)

obs.frand1 <- which(flrand == rand.filler[1])
obs.frand2 <- which(flrand == rand.filler[2])
obs.frand3 <- which(flrand == rand.filler[3])
obs.frand4 <- which(flrand == rand.filler[4])
obs.frand5 <- which(flrand == rand.filler[5])
obs.frand6 <- which(flrand == rand.filler[6])
obs.frand7 <- which(flrand == rand.filler[7])
obs.frand8 <- which(flrand == rand.filler[8])
obs.frand9 <- which(flrand == rand.filler[9])
obs.frand10 <- which(flrand == rand.filler[10])
obs.frand11 <- which(flrand == rand.filler[11])
obs.frand12 <- which(flrand == rand.filler[12])
obs.frand13 <- which(flrand == rand.filler[13])

if(length(obs.frand1) == 0){
obs.frand1 <- 0
```

```
}else
{
obs.frand1 <- min(obs.frand1):max(obs.frand1)
}

if(length(obs.frand2) == 0){
obs.frand2 <- 0
}else
{
obs.frand2 <- min(obs.frand2):max(obs.frand2)
}

if(length(obs.frand3) == 0){
obs.frand3 <- 0
}else
{
obs.frand3 <- min(obs.frand3):max(obs.frand3)
}

if(length(obs.frand4) == 0){
obs.frand4 <- 0
}else
{
obs.frand4 <- min(obs.frand4):max(obs.frand4)
}

if(length(obs.frand5) == 0){
obs.frand5 <- 0
}else
{
obs.frand5 <- min(obs.frand5):max(obs.frand5)
}

if(length(obs.frand6) == 0){
obs.frand6 <- 0
}else
{
obs.frand6 <- min(obs.frand6):max(obs.frand6)
}

if(length(obs.frand7) == 0){
obs.frand7 <- 0
}else
{
obs.frand7 <- min(obs.frand7):max(obs.frand7)
}

if(length(obs.frand8) == 0){
obs.frand8 <- 0
}else
{
obs.frand8 <- min(obs.frand8):max(obs.frand8)
}
```

```
if(length(obs.frand9) == 0){
obs.frand9 <- 0
}else
{
obs.frand9 <- min(obs.frand9):max(obs.frand9)
}

if(length(obs.frand10) == 0){
obs.frand10 <- 0
}else
{
obs.frand10 <- min(obs.frand10):max(obs.frand10)
}

if(length(obs.frand11) == 0){
obs.frand11 <- 0
}else
{
obs.frand11 <- min(obs.frand11):max(obs.frand11)
}

if(length(obs.frand12) == 0){
obs.frand12 <- 0
}else
{
obs.frand12 <- min(obs.frand12):max(obs.frand12)
}

if(length(obs.frand13) == 0){
obs.frand13 <- 0
}else
{
obs.frand13 <- min(obs.frand13):max(obs.frand13)
}

frand1 <- flrand[obs.frand1, ]
frand2 <- flrand[obs.frand2, ]
frand3 <- flrand[obs.frand3, ]
frand4 <- flrand[obs.frand4, ]
frand5 <- flrand[obs.frand5, ]
frand6 <- flrand[obs.frand6, ]
frand7 <- flrand[obs.frand7, ]
frand8 <- flrand[obs.frand8, ]
frand9 <- flrand[obs.frand9, ]
frand10 <- flrand[obs.frand10, ]
frand11 <- flrand[obs.frand11, ]
frand12 <- flrand[obs.frand12, ]
frand13 <- flrand[obs.frand13, ]

frand1 <- as.matrix(frand1)
frand2 <- as.matrix(frand2)
frand3 <- as.matrix(frand3)
frand4 <- as.matrix(frand4)
frand5 <- as.matrix(frand5)
```

```
frand6 <- as.matrix(frand6)
frand7 <- as.matrix(frand7)
frand8 <- as.matrix(frand8)
frand9 <- as.matrix(frand9)
frand10 <- as.matrix(frand10)
frand11 <- as.matrix(frand11)
frand12 <- as.matrix(frand12)
frand13 <- as.matrix(frand13)

frand <- rbind(frand1, frand2, frand3, frand4, frand5, frand6, frand7,frand8,
frand9, frand10, frand11, frand12, frand13)

c.frand <- cbind(frand, filler.per.1, filler.per.2, filler.per.3, filler.per.4,
filler.per.5, filler.per.6, filler.per.7, filler.per.8, filler.per.9,
filler.per.10, filler.per.11, filler.per.12, filler.per.13)

fordered <- c.frand[order(c.frand[, 1]), ]

obs.f1 <- which(fordered[ , 1] == frand.ordered[1])
obs.f2 <- which(fordered[ , 1] == frand.ordered[2])
obs.f3 <- which(fordered[ , 1] == frand.ordered[3])
obs.f4 <- which(fordered[ , 1] == frand.ordered[4])
obs.f5 <- which(fordered[ , 1] == frand.ordered[5])
obs.f6 <- which(fordered[ , 1] == frand.ordered[6])
obs.f7 <- which(fordered[ , 1] == frand.ordered[7])
obs.f8 <- which(fordered[ , 1] == frand.ordered[8])
obs.f9 <- which(fordered[ , 1] == frand.ordered[9])
obs.f10 <- which(fordered[ , 1] == frand.ordered[10])
obs.f11 <- which(fordered[ , 1] == frand.ordered[11])
obs.f12 <- which(fordered[ , 1] == frand.ordered[12])
obs.f13 <- which(fordered[ , 1] == frand.ordered[13])

if(length(obs.f1) == 0){
obs.f1 <- 0
}else
{
obs.f1 <- min(obs.f1):max(obs.f1)
}

if(length(obs.f2) == 0){
obs.f2 <- 0
}else
{
obs.f2 <- min(obs.f2):max(obs.f2)
}

if(length(obs.f3) == 0){
obs.f3 <- 0
}else
{
obs.f3 <- min(obs.f3):max(obs.f3)
}

if(length(obs.f4) == 0){
```

```
obs.f4 <- 0
}else
{
obs.f4 <- min(obs.f4):max(obs.f4)
}

if(length(obs.f5) == 0){
obs.f5 <- 0
}else
{
obs.f5 <- min(obs.f5):max(obs.f5)
}

if(length(obs.f6) == 0){
obs.f6 <- 0
}else
{
obs.f6 <- min(obs.f6):max(obs.f6)
}

if(length(obs.f7) == 0){
obs.f7 <- 0
}else
{
obs.f7 <- min(obs.f7):max(obs.f7)
}

if(length(obs.f8) == 0){
obs.f8 <- 0
}else
{
obs.f8 <- min(obs.f8):max(obs.f8)
}

if(length(obs.f9) == 0){
obs.f9 <- 0
}else
{
obs.f9 <- min(obs.f9):max(obs.f9)
}

if(length(obs.f10) == 0){
obs.f10 <- 0
}else
{
obs.f10 <- min(obs.f10):max(obs.f10)
}

if(length(obs.f11) == 0){
obs.f11 <- 0
}else
{
obs.f11 <- min(obs.f11):max(obs.f11)
}
```

```
if(length(obs.f12) == 0){
obs.f12 <- 0
}else
{
obs.f12 <- min(obs.f12):max(obs.f12)
}

if(length(obs.f13) == 0){
obs.frand13 <- 0
}else
{
obs.f13 <- min(obs.f13):max(obs.f13)
}


new.f1 <- fordered[obs.f1, 2]
new.f2 <- fordered[obs.f2, 3]
new.f3 <- fordered[obs.f3, 4]
new.f4 <- fordered[obs.f4, 5]
new.f5 <- fordered[obs.f5, 6]
new.f6 <- fordered[obs.f6, 7]
new.f7 <- fordered[obs.f7, 8]
new.f8 <- fordered[obs.f8, 9]
new.f9 <- fordered[obs.f9, 10]
new.f10 <- fordered[obs.f10, 11]
new.f11 <- fordered[obs.f11, 12]
new.f12 <- fordered[obs.f12, 13]
new.f13 <- fordered[obs.f13, 14]

if(length(new.f1) == 0){
f1.obs <- 0
}else
{
f1.obs <- fstart1:fend1
}

if(length(new.f2) == 0){
f2.obs <- 0
}else
{
f2.obs <- fstart2:fend2
}

if(length(new.f3) == 0){
f3.obs <- 0
}else
{
f3.obs <- fstart3:fend3
}

if(length(new.f4) == 0){
f4.obs <- 0
}else
```

```
{
f4.obs <- fstart4:fend4
}

if(length(new.f5) == 0){
f5.obs <- 0
}else
{
f5.obs <- fstart5:fend5
}

if(length(new.f6) == 0){
f6.obs <- 0
}else
{
f6.obs <- fstart6:fend6
}

if(length(new.f7) == 0){
f7.obs <- 0
}else
{
f7.obs <- fstart7:fend7
}

if(length(new.f8) == 0){
f8.obs <- 0
}else
{
f8.obs <- fstart8:fend8
}

if(length(new.f9) == 0){
f9.obs <- 0
}else
{
f9.obs <- fstart9:fend9
}

if(length(new.f10) == 0){
f10.obs <- 0
}else
{
f10.obs <- fstart10:fend10
}

if(length(new.f11) == 0){
f11.obs <- 0
}else
{
f11.obs <- fstart11:fend11
}

if(length(new.f12) == 0){
```

```
f12.obs <- 0
}else
{
f12.obs <- fstart12:fend12
}

if(length(new.f13) == 0){
f13.obs <- 0
}else
{
f13.obs <- fstart13:fend13
}

test1[f1.obs] <- new.f1
test1[f2.obs] <- new.f2
test1[f3.obs] <- new.f3
test1[f4.obs] <- new.f4
test1[f5.obs] <- new.f5
test1[f6.obs] <- new.f6
test1[f7.obs] <- new.f7
test1[f8.obs] <- new.f8
test1[f9.obs] <- new.f9
test1[f10.obs] <- new.f10
test1[f11.obs] <- new.f11
test1[f12.obs] <- new.f12
test1[f13.obs] <- new.f13

################## End of D-O Event Scattering ##################


#####################################################################
########### Cor for Block Bootstrap of GISP2 and EDML ###########
#####################################################################

all.cor.sim <- ccf(test1, edml$res, type = "correlation", plot = FALSE, lag.max
= 92)

sim.cor[, j] <- all.cor.sim$acf

######### Cor for Block Bootstrap of GISP2 and EDML ############


#####################################################################
###################### Statistic of Interest ####################
#####################################################################

max.lag.num <- which.max(sim.cor[, j])
max.lag <- rbind(max.lag, cbind(max.lag.num, sim.cor[max.lag.num, j]))

max.abs.num <- which.max(abs(sim.cor[, j]))
max.abs <- rbind(max.abs, cbind(max.abs.num, sim.cor[max.abs.num, j]))

cor.0 <- rbind(cor.0, sim.cor[93, j])
```

```
################## End of Statistic of Interest ###############
cat("iteration",j,"\n")
}


write.table(sim.cor, file = ".../all correlations.txt", row.names = FALSE,
col.names = FALSE)

write.table(max.lag, file = ".../max lag correlations.txt", row.names = FALSE,
col.names = c("lag", "corr"))

write.table(max.abs, file = ".../max abs correlations.txt", row.names = FALSE,
col.names = c("lag", "corr"))

write.table(cor.0, file = ".../cor 0 correlations.txt", row.names = FALSE,
col.names = FALSE)
```

C.3.2 Computing p-values

```
# Reading data in
gisp2 <- read.table(".../GISP2 47-20 kyr detrend interp.txt", header = TRUE)

edml <- read.table(".../EDML 47-20 kyr detrend interp.txt", header = TRUE)

all.cor <- ccf(gisp2$res, edml$res, type = "correlation", plot = FALSE, lag.max
= 92)
orig.cor <- cbind(all.cor$lag, all.cor$acf)
cor.0 <- orig.cor[93, 2]

numsim <- 10000

##################### Correlation of 0 ##########################

pval.0 <- function(sim.data){

num.obs <- which(sim.data >= cor.0)
num <- length(num.obs) + 1
p.val <- num/(numsim + 1)
p.val
}

# Resampled D-O events
bm1.r.0 <- read.table(".../cor 0 correlations.txt")

p.bm1.r.0 <- pval.0(bm1.r.0)

# Shuffled D-O events
bm.s.0 <- read.table(".../Shuffled D-O events/BM/Results/cor 0 correlations.txt")

p.bm.s.0 <- pval.0(bm.s.0)

##################### Correlation at lags ########################
numlags <- length(all.cor$acf)
```

```
cors <- all.cor$acf
pval.lag <- function(sim.data){
p.val <- NULL
for(i in 1:numlags){

num.obs <- which(abs(sim.data$corr) >= abs(cors[i]))
num <- length(num.obs) + 1
p.val.part <- num/(numsim + 1)
p.val <- rbind(p.val, p.val.part)
}
p.val
}

# Resampled D-O events
bm1.r.l <- read.table(".../max abs correlations.txt", header = TRUE)

pbm1.r.l <- pval.lag(bm1.r.l)

# Shuffled D-O events
bm.s.l <- read.table(".../Shuffled D-O events/BM/Results/max abs correlations.
txt", header = TRUE)

pbm.s.l <- pval.lag(bm.s.l)

# Graphically summarizing results
# Making a plot of the correlation and showing significance
cor.plot <- function(sim.data){

plot(1:185, orig.cor[, 2], type = "l", xlab = "Lag in Years", ylab =
"Correlation", axes = FALSE)
axis(1, at= c(1, 47, 93, 139, 185), lab = c("-2,500","-1,250", "0",
"1,250", "2,500"))
axis(2, at = c(-0.2, -0.1, 0.0, 0.1, 0.2, 0.3, 0.4), lab = c("-0.2",
"-0.1", "0.0", "0.1", "0.2", "0.3", "0.4"))
box(which = "plot", lty = "solid")

sig.pts <- which(sim.data <= 0.05)
polygon(c(sig.pts[1], sig.pts[length(sig.pts)], sig.pts[length(sig.pts)],
sig.pts[1]), c(-0.3, -0.3, 0.5, 0.5), col = "grey", border = "grey")

lines(1:185, orig.cor[, 2], lwd = 3)

}

cor.plot(pbm1.r.l)
cor.plot(pbm.s.l)

# Making a plot of the p-values
pval.plot <- function(sim.data, sim.cor0){
plot(1:185, sim.data, type = "l", lwd = 3, xlab = "Lag in Years", ylab =
"p-value", axes = FALSE)
axis(1, at= c(1, 47, 93, 139, 185), lab = c("-2,500", "-1,250", "0",
"1,250", "2,500"))
axis(2, at = c(0.0, 0.2, 0.4, 0.6, 0.8, 1.0), lab = c("0.0", "0.2",
```

```
"0.4", "0.6", "0.8", "1.0"))
box(which = "plot", lty = "solid")

abline(h = 0.05, lwd = 3, col = "red", lty = 2)
points(93, sim.cor0, lwd = 2, pch = 19)
}

pval.plot(pbm1.r.l, p.bm1.r.0)
pval.plot(pbm.s.l, p.bm.s.0)
```