

The utility of statistical analysis in structural geology

Nicolas M. Roberts^{1*}, Basil Tikoff¹, Joshua R. Davis², Tor Stetson-Lee¹

^{1*}*Department of Geoscience, University of Wisconsin—Madison. 1215 West Dayton St., Madison, WI 53715*

²*Department of Mathematics and Statistics, Department of Computer Science, Carleton College, Northfield, MN*

* nmroberts@wisc.edu

* (608) 262-4678

Key words: structural geology; directional statistics; orientation statistics; western Idaho shear zone; inference; regressions

Abstract

Recent advances in statistical methods for structural geology make it possible to treat nearly all types structural geology field data. These methods provide a way to objectively test hypotheses and to quantify uncertainty, and their adoption into standard practice is paramount to the future of structural geology. We provide a blueprint for structural geologists seeking to incorporate statistics into their workflow through statistical analyses in two locations within the western Idaho shear zone. In the N-S West Mountain location, we test the published interpretation that there is a bend in shear zone at the kilometer scale. Directional statistics on foliations corroborate this interpretation, while orientation statistics on foliation-lineation pairs do not. This discrepancy leads us to reconsider an assumption made in the earlier work. In the NW-SE Ahsahka location, we present results from a full statistical analysis of foliation-lineation pairs, including data visualization, regressions, and inference. These results agree with thermochronological

evidence that suggests that the Ahsahka segment comprises two distinct, subparallel shear zones. The R scripts that were used for both statistical analyses can be found in the appendices, with the intention that they be downloaded and run alongside the results section of this paper.

1. Introduction

Structural geologists routinely work with datasets that are logistically limited to small sample size and/or spatial extent. When working with such data, an important—but under-appreciated—task should be to determine what can reasonably be interpreted about the geologic system in question. This determination depends on the uncertainty that arises because the dataset is an incomplete representation of the larger system. The field of statistics is fundamentally concerned with this data-to-system uncertainty, and statistical methods have important utility for any empirical research. As structural geologists, we can use statistics to better identify trends, understand mean(s) and dispersion in datasets, test hypotheses, evaluate implicit assumptions, and communicate the confidence of our interpretations to peers.

In most publications, structural geologists make interpretations using quantitative data (e.g. fabric measurements) and qualitative estimates of uncertainty. The lack of statistical treatment of structural geology data is in part a historical issue: there is not a strong tradition of training structural geologists in statistics. As a discipline born out of field studies and geologic mapping, early structural geology methods—including quantitative ones—developed without a statistical framework. Even with the eventual development of such a framework for *directional* (rays and lines) data types such as paleomagnetic poles, lineation, pole to foliation, paleocurrent, and fault striations (e.g. Davis and Sampson, 1986; Ducharme et al., 1985; Fisher et al., 1989; Jupp and Mardia, 1989; Merrett and Allmendinger, 1990; Yonkee and Weil, 2015), the statistically savvy

structural geologist is still unusual. Though contouring directional data has become commonplace thanks to computer programs such as Stereonet (version 10.0.0; Allmendinger, 2017) and Orient (version 3.6.3; Vollmer, 2017), structural geologists do not generally make use of directional statistics to report statistical descriptors (mean, dispersion) or perform hypothesis tests in a statistically rigorous fashion.

Another reason structural geologists do not generally employ statistics is that many geologic data are not rays or lines, and thus cannot be treated with directional statistics. Until recently, there was no unified framework for the statistical treatment of *orientation* (line-in-plane) data like foliation-lineation pairs, fault planes with slickenlines, axial planes with hinges, or focal mechanisms. Davis and Titus (2017) have developed the mathematical background and theory of orientation statistics in order for it to be accessible by structural geologists. Moreover, they developed a free R programming language library for both direction and orientation statistics, called *geologyGeometry* (download at: <http://www.joshuadavis.us/software/>). Tools in the library include advanced plotting, regression algorithms, and parametric and non-parametric methods for inference, including hypothesis testing. The *geologyGeometry* library calls on other R libraries, including *Directional* for directional statistics (Tsagris and Athineou, 2016).

This contribution describes how to perform statistical analysis on structural geology data, and illustrates why incorporating statistics into a structural geology workflow is critical to the future of structural geology. Statistical analysis of two datasets from the western Idaho shear zone system, Idaho, USA are described in detail. The datasets were chosen because they address common questions in structural geology. A cursory geologic context is provided for each dataset (see Appendix 1 for additional details). The analysis of these two datasets reveals how thinking statistically leads to a more objective approach to interpretations and a quantified understanding

of the uncertainty surrounding these interpretations. By demonstrating both its methodology and utility, we hope to motivate the adoption of statistics into the standard structural geology workflow. All statistical analyses were done with the *geologyGeometry* R library, and the full analyses are shared in the appendices. Readers are encouraged to download the *geologyGeometry* R library and run the scripts found in the appendices line by line as they read. The scripts will output interactive, rotatable versions of many of the plots found in the figures.

2. The statistical approach

In the statistical analysis applied to each dataset in this paper, a new workflow motivated by both statistical protocol and geologic expertise has been applied. This workflow highlights two types of questions in statistics that are particularly relevant for structural geologists. First, are two datasets or subsets of a single dataset (e.g. two geographic domains) sampled from the same population? Second, are there real, systematic trends in the data, based on geographic position or any other variable?

Figure 1 presents a diagram of this workflow. First, the structural geologist visualizes the data in a variety of plots and maps. As a result of visualization, the geologist makes hypotheses and qualitative interpretations about the geologic system under study (lower path in Fig. 1), from which possible conceptual models and predictions are developed. Simultaneously, a statistical protocol is executed that is necessary for any dataset (upper path). Model predictions may be objectively tested by regressions (grey arrows), in which case the upper and lower path overlap. The data can be statistically described by mean and dispersion, and if there are no systematic spatial tendencies in the data, then inferences can be made about how well the population mean is known (the uncertainty of the population mean). The upper and lower path interact again when

model predictions are tested using this uncertainty in statistical hypothesis testing. A statistical hypothesis is formulated as a null hypothesis (e.g., “There is no difference in the mean of the populations from which dataset *A* and dataset *B* were sampled”) and an alternative hypothesis (e.g., “The means of the populations from which dataset *A* and dataset *B* were sampled do not have the same mean”). The null hypothesis is rejected or fails to be rejected based upon a credibility or confidence threshold, commonly 95%, or a *p*-value threshold, usually $p < 0.05$. A rejection of the null hypothesis leads a structural geologist to conclude that dataset *A* and dataset *B* are sampled from different populations. Importantly, though, the failure to reject the null hypothesis would *not* lead a structural geologist to conclude that dataset *A* and dataset *B* come from the same population—only that there is not strong evidence that they came from different populations.

Interpretations about the geologic system all pass through the statistical portion of the workflow, but importantly statistics are only useful insofar as they complement geologic expertise to develop statistically constrained conclusions about the geologic system. The first example in this paper focuses on the “Mean, dispersion,” “Inference about the mean,” and “Statistical hypothesis testing” boxes of the statistical workflow (Fig. 1) to statistically test a published geologic interpretation. The second example illustrates a path through the workflow (as shown by railroad ties on arrows in Fig. 1).

3. Background: direction and orientation data

The mathematical description of a geologic structure’s orientation depends on the type of geologic structure. A lineation can be described by two angles, a trend and plunge, or by a vector in Cartesian (north, east, up) coordinates. A foliation can also be described by two angles—a strike and a dip—and similarly can be described as a single Cartesian vector that defines the pole to the

foliation. Foliations and lineations are both examples of *directional* data, meaning that a single line or ray is sufficient to uniquely describe the geometry. A wealth of statistical techniques have been developed for directional data (e.g. Mardia and Jupp, 2000), termed *directional statistics*.

In contrast, a foliation-lineation pair is a line within a plane. At a minimum, three angles are required to describe a unique foliation-lineation: a strike, a dip, and a rake. Foliation-lineation pairs are an example of *orientation* data, and are treatable by *orientation statistics* (Davis and Titus, 2017). For statistical treatment, orientation data can be represented by a 3 by 3 rotation matrix, with the first row comprising the Cartesian vector of the pole to foliation, the second row comprising the vector of the lineation, and the third row comprising the vector that is orthogonal to the first two rows (Davis and Titus, 2017).

4. Application 1: The western Idaho shear zone near West Mountain, ID

The western Idaho shear zone forms a steep and abrupt north-south boundary between accreted terranes and the cratonic edge of the North American Cordillera (Armstrong et al., 1977; Fleck and Criss, 1985; Manduca et al., 1992; Fleck and Criss, 2004; Braudy et al., 2017). The ~5 kilometer wide shear zone is characterized by highly deformed orthogneisses. In the West Mountain area, the western Idaho shear zone is dextral, but sub-vertical lineations suggest transpression (Giorgis et al., 2008; Giorgis et al., 2016). The shear zone system bends at the 100-km scale to follow the cratonic boundary as defined by the $^{87}\text{Sr}/^{86}\text{Sr}$ isopleth.

A recent structural study suggests that a subtle bend in shear zone orientation can also be detected at the kilometer scale near West Mountain, ID. Braudy et al. (2017) collected a dataset of field fabrics including both foliation-only measurements as well as foliation-lineation pairs (Fig. 2). They plot both types of fabric data in equal area nets and interpret a ~20° rotation between

foliation strike in the North and South of the field area. In this section, we demonstrate a statistical analysis that is able to provide a more objective test of the interpretation of Braudy et al. (2017). In addition, we test the assumption that the foliations from the foliation-only dataset and the foliation-lineation dataset are the same. Foliation-only and foliation-lineation pairs are treated separately because they are different data types.

4.1 Directional statistics on foliation-only data

Foliation-only data comprise 148 field fabric measurements. Braudy et al. (2017) divide these data into three geographic domains: northern ($n = 56$), central ($n = 23$), and southern ($n = 69$) (Fig. 2B). For the sake of comparison, these domain divisions are used in this statistical analysis.

For each domain, the data are used to make an inference about the population mean. This calculation is done by bootstrapping (Efron and Tibshirani, 1994) and by applying three two-sample tests for comparison. Bootstrapping is repeated sampling with replacement. The implementation of the bootstrapping routine in the *geologyGeometry* R library is straightforward to use and automatically computes confidence regions (see Appendix 2 for a simplified description, or Davis and Titus (2017) for a full description). The result of bootstrapping is a cloud of means, whose center approximates the mean of the dataset and whose density at any given point is related to the likelihood of that point being the population mean. The 95% confidence ellipse of each domain is calculated using the Mahalanobis distance (Mahalanobis, 1936).

To determine whether the foliations in each domain come from different populations, a series of statistical tests are devised. The null hypotheses are that the population mean of one

domain (e.g. northern) is the population mean of another domain (e.g. southern). The null hypothesis is rejected if the 95% confidence regions of the two domains in question do not overlap.

Results of bootstrapping and 95% confidence region calculations are summarized in Figure 3A. The 95% confidence ellipses of the southern and central domains overlap, but neither of these domains overlap with the northern region. Therefore, we fail to reject the null hypothesis that the southern and central domains are sampled from the same population, while for comparisons with the northern domain, we reject the null hypothesis of a single population at 95% confidence.

In addition to bootstrapping, three types of two-sample tests were applied to each pair of domains. See Appendix 2 for a brief description of each test. A Wellner test (Wellner, 1979) yields a $p < 0.0001$ based on 10,000 permutations for the northern and southern domains as well as for the northern and central domains. The p -value for the southern and central comparison is 0.427. Two variations of Watson inference tests were performed on the data (Mardia and Jupp, 2000), one that assumes tightly concentrated data and the other that assumes large sample size. Respectively, these tests yielded p -values of 0.000001 and 0 (northern and southern domains), 0.00002 and 0.000005 (northern and central domains), and 0.233 and 0.131 (central and southern domains). These p -values agree well with the bootstrapping results, although some caution is advised, since these tests make assumptions about how the data are distributed. Taken together, these tests provide strong evidence against the null hypotheses that the northern domain and the central/southern domains are sampled from the same population.

Given the statistically significant difference between the northern domain and the other two domains, we calculate the rotational difference between the northern and southern domains. The axis and magnitude of rotation between the northern and southern domains is determined by computing the minimum rotations between 10,000 pairs of northern and southern bootstrap means.

Results from this analysis are summarized in Figure 3A. The mean rotation is $12.20^\circ \pm 3.8^\circ$ (2σ) with a mean rotation axis that trends 164.3° and plunges 68.8° . These results contrast the interpretation of Braudy et al. (2017), who suggested a $\sim 20^\circ$ rotation and implicitly assumed a vertical axis rotation.

4.2 Orientation statistics on foliation-lineation data

Foliation-lineation data comprise 129 field fabric measurements. The data are analyzed with the same geographic domains described above: northern ($n = 16$), central ($n = 34$), and southern ($n = 79$).

For each domain, both a bootstrapping method and a Markov chain Monte Carlo (MCMC) simulation (Davis and Titus, 2017) produce clouds of possible means from which a confidence region (for bootstrapping) or a credible region (for MCMC) can be computed. See Appendix 2 for a brief description. In general, for small sample sizes ($n < 30$), MCMC returns credible regions with accurate size, but the regions tend to be unrealistically isotropic. By contrast, bootstrapping returns more realistic anisotropic confidence regions, but the size of the region is consistently underestimated. Because of these complementary strengths and weaknesses, it is helpful to use both approaches.

The null hypotheses for foliation-lineation pairs are identical to those for foliations described previously. If the bootstrap/MCMC confidence/credible regions do not overlap, then the null hypothesis can be rejected at 95% confidence/credibility.

Figure 3B shows the results of both bootstrapping and MCMC. The null hypothesis that the northern and central domains are sampled from populations with the same mean cannot be rejected using MCMC, but can be rejected using bootstrapping at 95% confidence. The same is

true for the null hypothesis with respect to the northern and southern domains. The null hypothesis that the southern and central domains are sampled from populations with the same mean cannot be rejected at 95% confidence/credibility. Because the MCMC method tends to have more accurate credibility than bootstrapping confidence intervals, these analyses do not provide strong evidence that differences among the three domains are statistically significant, although there is weak evidence that the null hypothesis can be rejected for the northern domain with respect to the other two domains.

4.3 Comparing foliation-only and foliation-lineation data

The statistical analysis of foliation-only and foliation-lineation fabric data from Braudy et al. (2017) leads to two different interpretations. From the foliation-only data, a $12.55^\circ \pm 3.30^\circ$ (2σ) rotation between the southern/central and northern domains is inferred. From the foliation-lineation data, no such difference can be inferred with statistical significance. This discrepancy motivates a statistical comparison of these two datasets.

In a final comparison, only the foliations are used from the foliation-lineation data so that directional statistics can be applied to both datasets. The null hypothesis for each domain is that the foliations from the foliation-lineation dataset are sampled from the same population as those from the foliation-only dataset. A comparison of the bootstrapped mean cloud for each domain (Fig. 3C) shows that the null hypothesis can be rejected with 95% confidence for the central and southern domains, but is not clearly rejected for the northern domain. This result is unexpected because foliation-only data and foliation-lineation data were collected in the same field area (similar extent and spacing) and were assumed to be the same (Fig. 2A).

4.4 Summary

The statistical analysis of foliation and foliation-lineation data in the West Mountain area of the western Idaho shear zone allows for interpretations with quantitative evaluation of uncertainty. In addition, statistical comparison between foliation-only and foliation-lineation data reveals that a basic assumption about the two datasets—that the foliation and foliation-lineation datasets are being sampled from the same population—may not be valid.

The interpretation that Braudy et al. (2017) make with respect to foliation-only differences between the southern/central and northern domains is reasonable. Statistical analysis shows that a rotation is statistically significant. The magnitude of that rotation, however, is likely smaller than Braudy et al. (2017) interpreted, even though both this paper and the earlier study assume the smallest rotation is the geologically real one. Further, the axis of shortest rotation is not vertical, as implicitly presumed in Braudy et al. (2017), but plunges steeply to the south.

Statistical analysis of foliation-lineation pairs does not corroborate the counterclockwise rotation of fabric from south to north proposed by Braudy et al. (2017). The difference in fabric orientation among the domains is not statistically significant.

Braudy et al. (2017) do not interpret foliation-lineation pairs independently of the foliation-only data. By combining the datasets, they assume that foliations from the two datasets were sampled from the same population within each domain. Three statistical comparisons of the foliation data from the two datasets within each domain reject this assumption with 95% confidence for the central and southern domains. All three domains of the foliation-lineation foliations plot in the gap between the northern and southern domains of the foliation-only foliations (Fig. 3C). There are several possible explanations for this discrepancy which motivate future work. One possibility is that LS-tectonites may have a different orientation than S-tectonites because of

strain partitioning within the western Idaho shear zone. Another possibility is that rocks that had only foliations also have weaker fabric, which may account for the larger spread of data. Whatever the case, new scientific questions arise from the statistical analysis that would not have been asked in the absence of statistical methods.

5. Application 2: Ahsahka and Woodrat Mountain shear zones near Orofino, ID

The Ahsahka shear zone is in structural continuity with the western Idaho shear zone, about 200 km north of the West Mountain area. The Ahsahka shear zone occurs within a 90° bend of the western Idaho shear zone system (e.g., Lewis et al., 2014). The current interpretation is that there is an older, parallel Woodrat Mountain shear zone in cryptic contact with the northeast boundary of the Ahsahka shear zone (Lewis et al., 2014, Schmidt et al., 2016), but this interpretation is based upon relatively little geochronological data and remains controversial.

A recent dataset collected by Stetson-Lee (2015) comprises foliation-lineation measurements from areas on either side of the cryptic boundary between what is currently mapped as the Ahsahka and Woodrat Mountain shear zones near Orofino, Idaho (Fig. 4). There is a generally NW-striking foliation throughout the field area. Cooling $^{40}\text{Ar}/^{39}\text{Ar}$ ages on hornblende, biotite, and muscovite suggest that rocks on either side of the inferred boundary between the Ahsahka and Woodrat Mountain shear zones have a protracted thermal history, and record at least two distinct events (Davidson, 1990).

The goal of this statistical analysis is to assess whether structural data support the current interpretation of two shear zones. If geographic domains have statistically significant orientation differences, are these differences consistent with the current inferred boundary? This statistical analysis illustrates the proposed structural geology workflow (Fig. 1), with a particular emphasis

on the “statistical protocol” path. First, the data are visualized through a variety of plots. Second, the data are tested for geographic trends using regressions, and are split into geographic domains as a result. Third, the domains are statistically described with mean and dispersion. Finally, the domains are compared using hypothesis testing. The statistical tests are motivated and informed by data from the literature (maps, cooling ages) and the conceptual model for shear zone boundary that arises from them.

5.1 Statistical analysis using orientation statistics

The Orofino dataset comprises 69 foliation-lineation pairs in three geographic areas: Domain 1 ($n = 23$), domain 2 ($n = 14$), and domain 3 ($n = 32$) (Fig. 5). The division of the data in this way is consistent with current interpretations of geologic boundaries and will be statistically tested.

Initial plots contain all the data, not yet divided into geographic domains (Fig. 5A). Equal-area net and equal-volume plots with Kamb contours show that the foliation-lineation data have an approximately unimodal distribution in their orientation. However, coloring the data by geographic location reveals a non-random relationship between orientation and geography. For example, when the data are color-coded by northing, there are clear domains of yellows and reds (Fig. 5A). In map view, this geographic dependence is apparent; measurements in the south have a north to northwest lineation trend, while in the north, the lineations trend northeast to east.

Before splitting the data into domains, it is critical to know whether this geographic dependency is systematic (i.e. can be described by a continuous function) or whether there are discrete differences of orientation in different geographic domains. A series of 18 geodesic regressions help answer this question (Fig. 5b). Each of these regressions fits a geodesic curve to

the data as a function of an azimuth (e.g. northing). The maximum R^2 of a geodesic regression is 0.13 (for an azimuth of 30°). A kernel regression, which fits a more complex function to the data, of 30° azimuth has an R^2 value of 0.522.

These low R^2 values suggest that the geographic dependency observed in the equal area and equal volume plots is probably not systematic, and leads to the division of the data into multiple domains (Fig. 5C). The same plotting and regression analysis for each domain suggests that there is no strong geographic dependence (see Appendix 6).

Within each domain, the data are approximately unimodal and symmetric about that mode, so the mean is an appropriate summary statistic. We use the Fréchet mean which is the point that minimizes the Fréchet variance (Table 1). The dispersion of the data can be described using the matrix Fisher maximum likelihood estimation. This dispersion measure is not meaningful geologically, but is critical to selecting which inference method is most appropriate (Davis and Titus, 2017). In this case, MCMC simulation is the best behaved. As a check, bootstrapping has also been done.

MCMC and bootstrapping results for each domain are shown in Fig. 5D and 5E, with 95% credible/confidence ellipsoids. The null hypotheses are that each pair of domains are sampled from populations with the same mean. The credible/confidence regions of domain 2 and 3 overlap appreciably, while the credible/confidence region of domain 1 does not overlap with the other two. The null hypothesis that domains 2 and 3 are sampled from populations with the same mean cannot be rejected. The null hypothesis that domain 1 and domain 3 are sampled from populations with the same mean can be rejected with 95% credibility/confidence. The null hypothesis for domains 1 and 2 can also be rejected with 95% credibility/confidence.

5.2 Summary

There are three first-order conclusions that can be drawn from the statistical analysis of the foliation-lineation pairs near Orofino, ID. First, there are geographic domains; within each domain the data are roughly unimodal and symmetric, and apparent spatial dependencies have consistently low R^2 values. Second, the difference between orientations of domains 2 and 3 is not statistically significant. Third, domain 1 is significantly different from the other two domains.

These results are consistent with the mapped boundary between the Ahsahka and Woodrat Mountain shear zones as defined by cooling ages. Domains 2 and 3 are along strike of one another, and have been previously interpreted to be part of the Woodrat Mountain shear zone. Domain 1, which is across strike from the other two domains, has been interpreted to be part of the later Ahsahka shear zone. The presence of distinct orientations in rocks with different $^{40}\text{Ar}/^{39}\text{Ar}$ cooling ages, confirmed to be statistically significant in this analysis, provides further evidence that there were two distinct shear zones, now located adjacent to one another.

6. Discussion

In both the West Mountain and Orofino datasets, a workflow that incorporates statistical analysis leads to interpretations that are tested in an objective way with reported uncertainty—some of which would not likely not have been made otherwise. At West Mountain, we show that foliation-lineation pairs are not statistically distinguishable in different geographic areas, and that foliations in the foliation-only dataset are demonstrably different from foliations in the foliation-lineation dataset. In addition, we computed the magnitude of rotation between the northern and southern domains in a way that incorporates the uncertainty about the mean of each domain. In the Orofino area, we show that the difference in orientation on opposite ends of the Dworshak

Reservoir could not be accounted for by systematic spatial variations and that the difference in orientation between the southwest shore of the reservoir (domain 1) and the northeast shore (domains 2 and 3) is statistically significant. This division is consistent with other geologic data.

The statistical approach has tangible scientific benefits for structural geology data. Statistical methods help to quantitatively identify spatial tendencies in high-dimensional data. They also can be used to quickly compute basic statistical descriptors of the dataset and uncertainty about the population mean, which helps guide the visualization of data. The uncertainty of the population mean is used to reject (or fail to reject) geologic hypotheses posed as statistical hypotheses; using this approach to testing hypotheses, geologic interpretations come with quantifiable uncertainties which structural geologists can report in publications. The statistical approach also allows structural geologists to assess the validity of implicit assumptions using the same methods that are used to test geologic hypotheses.

6.1 Identifying spatial tendencies

In our regressions of foliation-lineation data, we treat each data point holistically as a rotation matrix. This approach is an improvement on standard practice in structural geology. The investigation of geographic trends in structural geology data usually involves the decomposition of high-dimensional data like foliation-lineation pairs into one-dimensional elements. For example, it is common to plot the strike of foliation against the distance from a shear zone, even though each data point is a strike, dip, and rake. These two-dimensional charts have some utility, but provide an incomplete view of each data point. Best-fit lines and associated R^2 values in these charts are problematic because such regressions should be informed by the other dimensions that comprise each data point. This partial view of the data can lead to false correlations or can fail to

368 reveal correlations entirely. By treating the data holistically during statistical analysis, structural
369 geologists can be more accurate in their identification of spatial dependencies.

370 Performing regressions is also a critical step when testing that the data are spatially
371 independent. Common techniques for inference about the mean of a population assume this
372 independence. An iterative process of visual inspection (plotting), regression analysis, and division
373 of the data into domains (as performed in the Orofino example) ensures that this assumption is
374 reasonable.

375 376 *6.2 Uncertainty about the mean*

377 Structural geology datasets are often relatively small and dispersed. This situation is
378 especially true for field datasets such as the ones described in this paper. Equal-area projection
379 computer programs widely used by structural geologists have built-in measures of mean and
380 dispersion for directional data (e.g. foliations only), but do not yet treat orientation data (e.g.
381 foliation-lineation pairs). For foliation-lineation data, we employ one of two equally valid
382 conceptions of the mean (Davis and Titus, 2017). To quantify how well the mean of the dataset
383 reflects the mean of the population from which the dataset was sampled, bootstrapping and MCMC
384 simulations produce clouds of means from which confidence and credibility regions can be
385 inferred. The confidence/credible region for the mean of a structural geology dataset has two main
386 functions. First, it contextualizes the mean of the dataset—a mean is not particularly useful if the
387 uncertainty about that mean is very large. Second, confidence/credible regions enable comparison
388 with other datasets using hypothesis testing.

389 390 *6.3 Hypothesis testing*

In both examples provided in this paper, an experienced structural geologist would most likely notice differences among some of the domains. Taking a statistical approach, structural geologists can test hypothesized differences in an objective way. In practice, if the 95% confidence/credible regions of two domains do not overlap, the null hypothesis that they are sampled from the same population can be rejected.

Statistical significance is especially important when the difference between two datasets is small or data are dispersed. In the West Mountain example, foliation-only foliations and foliation-lineation foliations plot in the same general area of the equal area net. While visual inspection may lead a structural geologist to suspect a difference between the two datasets, it is only through statistical hypothesis testing that the geologist can say with 95% confidence that this difference is not due to random variation within the same population. We are able to rely on this interpretation to ask further questions—such as why foliation-lineation pairs are different from foliation-only data—precisely because we have rejected the null hypothesis that they come from the same population.

6.4 Using hypothesis testing and regressions to assess assumptions

A statistical comparison of the two West Mountain datasets led to the conclusion that foliations from foliation-only data were likely not sampled from the same population as those from foliation-lineation data. This finding leads us to reject an assumption that Braudy et al. (2017) made which seemed logical. The ability to quickly assess such assumptions is a key advantage of the statistical workflow. In part, this advantage comes from a shift in perspective, because the statistical approach forces the articulation (and thus awareness) of the assumptions we make when analyzing data.

414

415 6.5 Better science through statistics

416 The use of statistics in structural geology may seem onerous, simply another task to
417 complete prior to submitting a manuscript. However, given the examples above we suggest that
418 there are many reasons to adopt this methodology. Taken together, the benefits of the statistical
419 approach make it easier to have the scientific integrity Feynman (1974) discussed in his famous
420 essay “Cargo Cult Science”:

421

422 *The first principle is that you must not fool yourself—and you are the easiest person to*
423 *fool. So you have to be very careful about that. After you've not fooled yourself, it's easy*
424 *not to fool other scientists. You just have to be honest in a conventional way after that.*

425

426 When conducting fieldwork, many (if not most) of the hypotheses that we initially formulate are
427 ultimately incorrect. The successful execution of science is the ability to generate and discard
428 hypotheses with relative efficiency. Statistics aids in perhaps the most difficult part of the scientific
429 process: exactly when to discard a hypothesis. Statistical analysis is used by most scientific
430 communities—including field scientists such as ecologists—to facilitate this process. As Henry
431 Pollack puts it in *Uncertain Science, Uncertain World*, the role of science is to separate the
432 “demonstrably false from the probably true” (Pollack, 2003), and the uncertainty computed by
433 statistical methods is a primary tool to fulfill that role. The relatively small size and large dispersion
434 common to structural geology datasets may seem a good excuse not to use statistics. In fact, these
435 characteristics are particularly compelling reasons to incorporate statistical tools into the structural

geology workflow. It can be tempting to over-interpret small datasets, and statistics provides a check on what interpretations are permissible given the small sample size.

Finally, the institution of science depends on the presentation of data and interpretations to scientific peers. Most data in structural geology papers present data in the form of stereonet or other representative documentation, neither of which allows other structural geologists to evaluate or use the dataset effectively. A clear statement of the tested hypotheses and the results would be a useful way to communicate the uncertainty of the data with respect to a specific model. The theory for the types of data that we collect has been addressed by Davis and Titus (2017) and the tools to statistically analyze data are now easily available. With the addition of specific use cases—as introduced in this contribution—we hope that both the methodology and its utility will be clear and accessible to the structural geology community.

Conclusion

Structural geology, especially structural geology in the field, is a science that benefits from the incorporation of statistical procedures. Field datasets are commonly small, geographically dispersed, and limited to small areas of good outcrop. Further, structural geology data are inherently high-dimensional, meaning that traditional ways of viewing data provide incomplete pictures of the data. The analysis of structural geology data within a statistical framework provides a way for structural geologists to more quantitatively understand and interrogate their data.

In this contribution, two typical structural geology field datasets were analyzed using direction and orientation statistics. In both cases, we employed a workflow in which geologic expertise interacts with statistical protocol to motivate geologically relevant statistical tests (Fig. 1). In this framework, statistics connects the collected dataset to the geologic system through

quantitative measures of uncertainty. We find significant utility in adopting such a workflow, particularly for datasets that are small and disperse. The statistical approach allowed us to interpret subtle differences in domains as real through hypothesis testing.

Statistical tools are critical to the future of structural geology. As structural geology datasets become available in open source databases, these statistical tools will be increasingly important. When combining datasets collected by different geologists over the same geographic extent, these tools provide a way to test whether combining datasets is permissible. When examining the same type of geologic feature at thousands of field locations worldwide, these tools provide a way to quantitatively compare geometries.

Acknowledgements

This work was supported by funding from the National Science Foundation EAR 1639748 to B. Tikoff and J. Newman. Thank you to V. Chatzaras, N. Garibaldi, R. Williams, A. Jones, C. Bate and the rest of the Structure and Tectonics group at UW-Madison for insightful manuscript comments.

References

- Allmendinger, R. W. (2017), Stereonet 10.0.0 [Computer software]. Retrieved from <http://www.geo.cornell.edu/geology/faculty/RWA/programs/stereonet.html>
- Armstrong, R. L., W. H. Taubeneck, and P. O. Hales (1977), Rb-Sr and K-Ar geochronometry of Mesozoic granitic rocks and their Sr isotopic composition, Oregon, Washington, and Idaho, *Geological Society of America Bulletin*, 88(3), 397–411.
[https://doi.org/10.1130/0016-7606\(1977\)88<397:RAKGOM>2.0.CO;2](https://doi.org/10.1130/0016-7606(1977)88<397:RAKGOM>2.0.CO;2)
- Benford, B., J. Crowley, M. Schmitz, C. J. Northrup, and B. Tikoff (2010), Mesozoic magmatism and deformation in the northern Owyhee Mountains, Idaho: Implications for along-zone variations for the western Idaho shear zone, *Lithosphere*, 2(2), 93–118.
<https://doi.org/10.1130/L76.1>
- Braudy, N., R. M. Gaschnig, D. Wilford, J. D. Vervoort, C. L. Nelson, C. Davidson, M. J. Kahn, and B. Tikoff (2017), Timing and deformation conditions of the western Idaho shear zone, West Mountain, west-central Idaho, *Lithosphere*, 9(2), 157–183.
<https://doi.org/10.1130/L519.1>
- Davidson, G. F. (1990), Cretaceous tectonic history along the Salmon River suture zone near Orofino, Idaho: Metamorphic structural and $^{40}\text{Ar}/^{39}\text{Ar}$ thermochronologic constraints (M.Sc. thesis). Location: Oregon State University.
- Davis, J. C., and R. J. Sampson (1986), *Statistics and data analysis in geology*, Wiley New York et al.

495 Davis, J. R., and S. J. Titus (2017), Modern methods of analysis for three-dimensional
 496 orientational data, *Journal of Structural Geology*, 96, 65–89.
 497 <https://doi.org/10.1016/j.jsg.2017.01.002>

498 Ducharme, G. R. (1985), Bootstrap confidence cones for directional data, *Biometrika*, 72(3),
 499 637–645. <https://doi.org/10.1093/biomet/72.3.637>

500 Feynman, R. P. (1974), Cargo cult science, *Engineering and Science*, 37(7), 10–13.

501 Fisher, N. I., and P. Hall (1989), Bootstrap confidence regions for directional data, *Journal of the*
 502 *American Statistical Association*, 84(408), 996–1002.
 503 <https://doi.org/10.1080/01621459.1989.10478864>

504 Fleck, R. J., and R. E. Criss (1985), Strontium and oxygen isotopic variations in Mesozoic and
 505 Tertiary plutons of central Idaho, *Contributions to Mineralogy and Petrology*, 90(2–3),
 506 291–308. <https://doi.org/10.1007/BF00378269>

507 Fleck, R. J., and R. E. Criss (2004), Location, age, and tectonic significance of the Western
 508 Idaho Suture Zone (WISZ). *Report: USGS Numbered Series*, 2004-1039, p. 48.

509 Giorgis, S., W. McClelland, A. Fayon, B. S. Singer, and B. Tikoff (2008), Timing of
 510 deformation and exhumation in the western Idaho shear zone, McCall, Idaho, *Geological*
 511 *Society of America Bulletin*, 120(9–10), 1119–1133. <https://doi.org/10.1130/B26291.1>

512 Giorgis, S., Z. D. Michels, L. Dair, N. Braudy, and B. Tikoff (2017), Kinematic and vorticity
 513 analyses of the western Idaho shear zone, USA, *Lithosphere*, 9(2), 223–234.
 514 <https://doi.org/10.1130/L518.1>

515 Jupp, P. E., and K. V Mardia (1989), A unified view of the theory of directional statistics, 1975-
 516 1988, *International Statistical Review/Revue Internationale de Statistique*, 57(3), 261–
 517 294. <https://doi.org/10.2307/1403799>

518 Lewis, R. S., K. L. Schmidt, R. M. Gaschnig, T. A. LaMaskin, K. Lund, K. D. Gray, B. Tikoff,
 519 T. Stetson-Lee, and N. Moore (2014), Hells Canyon to the Bitterroot front: A transect
 520 from the accretionary margin eastward across the Idaho batholith, *Geological Society of*
 521 *America Field Guides*, 37, 1–50.

522 Mahalanobis, P. C. (1936), On the generalized distance in statistics, *Proceedings of the National*
 523 *Institute of Sciences of India*, 1936, 49–55.

524 Manduca, C. A., L. T. Silver, and H. P. Taylor (1992), $^{87}\text{Sr}/^{86}\text{Sr}$ and $^{18}\text{O}/^{16}\text{O}$ isotopic systematics
 525 and geochemistry of granitoid plutons across a steeply-dipping boundary between
 526 contrasting lithospheric blocks in western Idaho, *Contributions to Mineralogy and*
 527 *Petrology*, 109(3), 355–372. <https://doi.org/10.1007/BF00283324>

528 Mardia, K. V, and P. E. Jupp (2000), *Directional statistics*, John Wiley & Sons.

529 Marrett, R., and R. W. Allmendinger (1990), Kinematic analysis of fault-slip data, *Journal of*
 530 *structural geology*, 12(8), 973–986. [https://doi.org/10.1016/0191-8141\(90\)90093-E](https://doi.org/10.1016/0191-8141(90)90093-E)

531 McClelland, W. C., and J. S. Oldow (2004), Displacement transfer between thick-and thin-
 532 skinned décollement systems in the central North American Cordillera, *Geological*
 533 *Society, London, Special Publications*, 227(1), 177–195.
 534 <https://doi.org/10.1144/GSL.SP.2004.227.01.10>

535 Pollack, H. N. (2003), *Uncertain science... uncertain world*, Cambridge University Press.

536 Rember, W. C., & Bennett, E. H. (1979). Geologic map of the Idaho Falls quadrangle. *Idaho,*
537 *Idaho Bureau of Mines and Geology Geologic Map Series, Idaho Falls, 2.*

538 Schmidt, K. L., R. S. Lewis, J. D. Vervoort, T. A. Stetson-Lee, Z. D. Michels, and B. Tikoff
539 (2017), Tectonic evolution of the Syringa embayment in the central North American
540 Cordilleran accretionary boundary, *Lithosphere*, 9(2), 184–204.
541 <https://doi.org/10.1130/L545.1>

542 Stetson-Lee, T. A. (2015), Using Kinematics and Orientational Statistics to Interpret
543 Deformational Events: Separating the Ahsahka and Dent Shear Zones Near Orofino, ID
544 (M.Sc. thesis). Location: University of Wisconsin--Madison.

545 Tsagris, M., and Athineou, G. (2016). Directional: Directional Statistics. R package version
546 1.8. <http://CRAN.R-project.org/package=Directional>

547 Vollmer, F. W. (2017), Software for the quantification, error analysis, and visualization of strain
548 and fold geometry in undergraduate field and structural geology laboratory experiences,
549 in *Geological Society of America Abstracts with Programs*, 49(2).
550 <https://doi.org/10.1130/abs/2017NE-291027>

551 Wellner, J. A. (1979), Permutation tests for directional data, *The Annals of Statistics*, 7(5), 929–
552 943. <http://www.jstor.org/stable/2958664>

553 Yonkee, W. A., and A. B. Weil (2015), Tectonic evolution of the Sevier and Laramide belts
554 within the North American Cordillera orogenic system, *Earth-Science Reviews*, 150,
555 531–593. <https://doi.org/10.1016/j.earscirev.2015.08.001>

557 **Figure Captions**

Figure 1. Schematic diagram of a structural geology workflow that takes advantage of statistical tools to aid interpretations of the geologic system. The grey box surrounds the statistical component of the workflow, and is a simplification of the statistical flowchart from Davis and Titus (2017). Railroad ties on arrows indicate portions of the workflow used in the examples in this paper. The structural geologist begins with an incomplete representation of the geologic system (the dataset). After visualizing the data, two simultaneous processes begin—the generation of geologic hypotheses/associated predictive models and a statistical protocol that should be done on any dataset. The statistical protocol interacts when testing the geologic hypotheses either through regressions or statistical hypothesis testing. Importantly, all interpretations of the geologic system run through the grey statistical box.

Figure 2. Simplified geologic map and overview of data from the West Mountain, ID area of the Late Cretaceous western Idaho shear zone published in Braudy et al. (2017). **A)** Geologic units of the western Idaho shear zone (Red—Muir Creek orthogneiss, Purple—Sage Hen orthogneiss, Magenta—Payette River Tonalite). The Muir Creek orthogneiss was the focus of the structural study in Braudy et al. (2017). **B)** Geographic locations and symbols of foliation-lineation data (left) and foliation-only data (right). There are 148 foliation-only measurements and 129 foliation-lineation pairs. **C)** Equal area nets with data for foliation-only (above) and foliation-lineation datasets (below), color-coded by the geographic domains used by Braudy et al. (2017) (Red—northern, Green—central, Blue—southern). Map modified from Braudy et al. (2017).

Figure 3. Summary of the statistical analyses for the West Mountain field fabrics dataset. **A)** An analysis of the claim from Braudy et al. (2017) that there is a 20° rotation between the northern and southern domains: Top, a lower hemisphere equal area projection (with zoomed-in cutout) with the 95% confidence regions for the mean of foliation-only data in each of the three domains (Red—northern, Green—central, Blue—southern) as determined from bootstrapping; Middle, a histogram of angular distances between bootstrap iterations of the northern and southern domains; Bottom, a visualization of the rotation computed from the bootstrapped angular distance and corresponding rotation axes. **B)** A series of two-sample hypothesis tests plotted on equal volume plots (with zoomed-in cutouts). Both bootstrapping and 95% confidence ellipsoids as well as Markov chain Monte Carlo (MCMC) mean probability clouds and their 95% credible ellipsoids are used to compare each pair of domains (Black—northern, Orange—central, Blue—southern). **C)** A comparison of 95% confidence ellipses from bootstrapping foliations. Foliations from foliation-lineation data are compared with those from foliation-only data within each domain: Colors are the same as in (A).

Figure 4. Simplified geologic map of the Orofino area, with the foliation-lineation dataset superimposed. Exposure of sheared Late Cretaceous basement below the Miocene Columbia River basalts is limited to the shoreline of Dvorshak reservoir. An interpretation of the boundary between the Woodrat Mountain and Ahsahka shear zones is shown. Modified from Rember and Bennett (1979).

Figure 5. Summary of statistical analysis for the Orofino, ID area foliation-lineation dataset. **A)** Two different plots of the foliation-lineation data colored by kilometers north: Left, an equal-area plot with lineations (squares) and foliation poles (circles), each with 2σ , 6σ , 10σ , 14σ , and 18σ Kamb contours; Right, an equal volume plot after Davis and Titus (2017) with translucent 2σ Kamb contours. Each point in the equal volume plot is a foliation-lineation pair represented as a rotation from a reference plane-line pair. Note that there are four copies of the dataset due to four-fold symmetry of such data (See Davis and Titus (2017) for more information). **B)** A series of 18 geodesic regressions testing geographic variation along specific azimuths. Each solid dot is a regression with a corresponding p -value (open circle). **C)** The geologic map from Figure 5 superimposed with the domains used in this statistical analysis. **D)** A series of two-sample hypothesis tests plotted on equal volume plots (with zoomed-in cutouts). MCMC mean probability clouds and their 95% credible regions as well as bootstrapped mean clouds and their 95% confidence region are used to compare each pair of domains (Black—domain 1, Orange—domain 2, Blue—domain 3). **E)** A lower-hemisphere, equal-area projection showing the results of the MCMC analysis. Colors are the same as for (D).

Table 1. The Fréchet mean strike, dip, and rake for the three domains in the Ahsahka segment of the western Idaho shear zone. Strike/dip/Rake are in right hand rule.

APPENDIX I: Geologic Background

Both datasets examined in this paper come from the western Idaho–Ahsahka shear zone system, Idaho, which forms a steep and abrupt boundary between the accreted terranes (consisting of collided magmatic arcs) and the cratonic edge of the North American Cordillera. (e.g., Armstrong et al., 1977; Fleck and Criss, 1985; Manduca et al., 1992; Fleck and Criss, 2004; Braudy et al., 2017) and recent seismic data (Stanciu et al., 2016; Davenport et al., 2017). Schmidt et al. (2016) noted that the timing of deformation is nearly identical within the western Idaho and Ahsahka shear zones, and they are in structural continuity.

The first dataset comes from the western Idaho shear zone portion of the boundary, near West Mountain, ID. In this segment, sub-vertical fabric is oriented ~ 020 in southwestern Idaho (Benford et al., 2010) and 000 in central Idaho (Manduca et al., 1993; Giorgis et al., 2008). The western Idaho shear zone is ~ 5 km wide and is characterized by highly deformed orthogneisses. Deformation is interpreted as dextral transpressional, with a vertical vorticity axis parallel to the lineation orientation (e.g., Giorgis et al., 2008; Michels et al., 2015) and an estimated angle of oblique convergence of $45\text{--}60^\circ$ (Giorgis et al., 2016). At West Mountain, metamorphosed sedimentary wall rocks involved in the shear zone record pressures of 4.5 kbar and temperatures of $\sim 730^\circ\text{C}$ (Braudy et al., 2017). Deformation occurred in the middle Cretaceous, constrained regionally to have occurred between 104–90 Ma with peak metamorphism at 100–97 Ma (Braudy et al., 2017). Of particular relevance to this study is that Braudy et al. (2017) noted a change in foliation orientation in the West Mountain segment, which they interpreted as reflecting a primary along-strike variation in the western Idaho shear zone.

The second dataset comes from the Ahsahka shear zone portion of the boundary. The Ahsahka shear zone occurs in a 90° change in orientation in the fabric of the zone near Orofino,

ID (e.g., Lewis et al., 2014). The NS orientation of the WISZ rotates to an EW orientation in the Ahsahka shear zone.

The exact nature of deformation around Orofino, ID, is confusing because multiple shear zones are located in the area and abundant flows of the Miocene Columbia River basalt group cover the region. Both Strayer et al. (1989) and Davidson (1990) recognized zones of ductilely deformed rocks and determined top-to-the-south/southwest shear sense indicators. These fabrics typically occur in mylonitic gneisses that display moderately north-northeast dipping foliation and steeply pitching lineation. Schmidt et al. (2016) designated these fabrics to be part of the Ahsahka shear zone, which is distinct from the older Woodrat Mountain shear zone (Lewis et al., 2014) to the northeast. Deformation in the Ahsahka shear zone occurred between 116-89 Ma (Schmidt et al., 2016), with dominantly reverse-sense shear and a horizontal vorticity vector perpendicular to the lineation direction (Giorgis et al., 2017). Critical for this study is that there are multiple shear zones in the Orofino area, including the older Woodrat shear zone (McClelland and Oldow, 2004, 2007; Lund et al., 2007).

APPENDIX 2: A primer on the inference methods used in this paper

Regressions

Geographic gradients are commonly observed in structural geology. Two common examples are strain gradients or change in orientation due to gentle regional folding. The observation of these gradients plays a major role in how we interpret data. Regressions are designed to detect and quantify the strength of these gradients.

In elementary statistics, the simplest regression takes the form of a best fit line, which minimizes the square of the distances between the data points and that line. The analogously simple

regression for orientation data is a *geodesic regression* (Davis and Titus, 2017). In our statistical analysis, a best-fit geodesic curve describes the steady-state relationship between orientation of fabric and an azimuthal direction. For example, if fabric rotates about a specific axis 5° every 100 meters to the northwest (a 45° azimuth), then a geodesic curve would accurately describe the geographic gradient. The R^2 of the regression provides a description of how closely the data fit the geodesic function. An R^2 of 1 represents a perfect fit, and an R^2 of 0 represents no fit.

3.2 Inference methods

Inference methods use the dataset to test hypotheses about the population. In this paper, we focus on two approaches to inference. When dealing with directional data like foliations, we use *bootstrapping* in addition to three more standard directional statistical tests. When dealing with orientation data like foliation-lineation pairs, we use both *bootstrapping* and a *Markov chain Monte Carlo* simulation.

Bootstrapping is non-parametric, meaning that it does not depend on an assumption of distribution (Efron and Tibshirani, 1994). Simply put, bootstrapping is repeated resampling with replacement. For example, if there are 20 foliation measurements, then each iteration of the bootstrapping routine selects 20 measurements from the dataset, but some of the original data are picked more than once and others not at all. We use 10,000 iterations. The mean of each of these perturbed iterations is recorded. When plotted, the density of the cloud of bootstrapped means reflects the sampling distribution of the mean, which is the concept that underlies confidence regions. The 95% confidence region around the mean of the dataset describes the uncertainty in the sample mean as an estimate of the population mean. It is computed by using principal

component analysis and the Mahalanobis distance (Mahalanobis, 1936) to fit an ellipsoid that contains 95% of the bootstrapped means.

The *Markov chain Monte Carlo* simulation takes an entirely different, probabilistic approach to quantifying uncertainty about the population mean. The method is a Bayesian approach; instead of asking how probable the dataset is given the statistical model (e.g. how well the data fit a normal distribution with a specific mean and variance), the Bayesian approach asks the opposite question: How probable a statistical model is given the dataset? The question posed in this way is called the *posterior probability*. The *Markov chain Monte Carlo* algorithm produces a set of iteratively perturbed statistical models which converge on the statistical model that best fits the data. In each step of the algorithm, the parameters of the statistical model from the previous step are modified slightly, and the posterior probability is calculated. If the posterior probability is better than it was for the previous step, then the new parameters are kept. Otherwise, the new step has a chance of being overwritten by the previous step. The value of the mean in each of these perturbed statistical models creates a list of means similar to bootstrapping, where the density of the means when plotted reflects the relative probability that a point represents the true population mean. A region containing 95% of the means is computed using Mahalanobis distance as above. This region is called a 95% *credible* region in recognition of its Bayesian origin. In practice we use credible regions and confidence regions similarly.

We use both bootstrapping and MCMC simulation because they each have advantages and weaknesses. In general, for small sample sizes ($n < 30$), MCMC returns credible regions with accurate size, but the regions tend to be unrealistically isotropic. By contrast, bootstrapping returns more realistically proportioned confidence regions, but the size of the region is consistently underestimated.

For foliations (directional data), we also employ three two-sample tests from the literature as a check on the bootstrap methodology. The Wellner two-sample test (Wellner, 1979) computes a T -statistic that provides a measure of how different the two samples are. The samples are then repeatedly scrambled with one another, and the T -statistic is calculated for each iteration. The percentage of T -statistics that are greater than the T -statistic of the original datasets (instances where scrambled datasets were more dissimilar than the original) serves as a p -value. The other two-sample tests are versions of the Watson two-sample test, in which an F statistic is computed and plotted on an F distribution for comparison with a $p = 0.05$ threshold value. The two Watson two-sample tests make different assumptions to approximate the F statistic; one assumes that the dataset is large, the other assumes that the data are tightly concentrated (Mardia and Jupp, 2000).

3.3 The null hypothesis

The inference methods described above are used to test *null hypotheses*. A null hypothesis is a statement that can be rejected (or fails to be rejected) based on inference. For example, a null hypothesis for a foliation dataset might be “the mean of the population is a foliation that strikes 077 and dips 35.” If 077/35 is outside of the 95% confidence bootstrap region, then we can reject the null hypothesis at 95% confidence. If, however, 077/35 falls within the 95% confidence region, we cannot reject the null hypothesis. Note that in the case that we fail to reject the null hypothesis, we *do not accept* it. Failure to reject the null hypothesis is an inconclusive result.

For the most part, this paper uses confidence/credible regions for testing the null hypothesis, because both bootstrapping and Markov chain Monte Carlo approaches result in confidence/credible regions, and null hypotheses are rejected or not based on whether or not the hypothesized value lies within these regions. We reject the null hypothesis when the confidence is

95% or more. The Wellner and Watson two-sample tests report p -values. We reject the null hypothesis when $p < 0.05$.

APPENDIX 3: West Mountain data

Appendices 3, 4, 5, and 6 are data files and R scripts used for the statistical analysis. Follow the directions below to access and run the scripts and data files in their proper format.

1. Go to <http://www.joshuadavis.us/software/> and download the *geologyGeometry* R library as well as the zip file containing:
 - a. *JSG_statsFunctions.r*
 - b. *JSG_statisticalAnalysis_WestMountain.r*
 - c. *JSG_statisticalAnalysis_Orofino.r*
 - d. *Fols_WestMt.csv*
 - e. *Follins_WestMt.csv*
 - f. *Follins_Ahs.csv*
2. Follow the instructions in section 2 of the *readme.pdf* within the *geologyGeometry* folder. The installation of R, R-studio, and required R packages is described.
3. Move the *geologyGeometry* folder out of the Downloads folder to a location of your choosing and save the three R scripts (*.r*) into the *geologyGeometry* folder.
4. Save the three datasets (*.csv*) in the *data* folder within *geologyGeometry*
5. At the top of each script, there is a line `setwd("~/Desktop/20170620geologyGeometry")` Change the text within the quote to match the path on your computer of the *geologyGeometry* folder.

763

764 *West Mountain Foliations*

location	easting	northing	strike	dip
10B352	562899	4915537	0	81
12b6	564099	4925419	0	76
12b9	565249	4925632	0	84
12b9	565249	4925632	0	80
12b10	565508	4925751	0	87
12b13	565492	4925823	0	80
12b19	563193	4920861	0	80
12b28	563550	4912053	0	75
12b73	564367	4925408	0	82
12b74	564609	4925466	0	83
12b9	565249	4925632	1	84
12b6	564099	4925419	2	80
12b9	565249	4925632	2	88
12b73	564367	4925408	2	88
12b74	564609	4925466	2	80
12b6	564099	4925419	3	75
12b6	564099	4925419	3	80
12b7	564258	4925370	3	83
12b9	565249	4925632	3	87
10B347	561990	4916474	4	76
12b63	564555	4925819	4	82
12b73	564367	4925408	4	79
186	561654	4911123	5	81
12b7	564258	4925370	5	73
12b7	564258	4925370	5	84
12b9	565249	4925632	5	81
12b9	565249	4925632	5	82
12b9	565249	4925632	5	90
12b73	564367	4925408	5	82
12b6	564099	4925419	6	79
12b6	564099	4925419	6	77
12b6	564099	4925419	6	81
12b7	564258	4925370	6	81
12b7	564258	4925370	6	85
12b9	565249	4925632	6	85
10b18	563252	4912347	7	77

10B340	562668	4916013	8	81
184	562273	4911342	8	68
12b6	564099	4925419	8	77
12b6	564099	4925419	8	81
12b6	564099	4925419	8	80
12b18	563126	4920162	8	84
12b74	564609	4925466	8	80
12b6	564099	4925419	9	75
12b7	564258	4925370	9	80
12b7	564258	4925370	9	80
12b49	561431	4922977	9	76
12b52	563344	4924630	9	78
10B260	564573	4916371	10	75
10B298	563985	4917688	10	68
10B317	562561	4919112	10	76
342	563437	4922678	10	85
12b6	564099	4925419	10	85
12b6	564099	4925419	10	72
12b9	565249	4925632	10	89
12b54	560198	4915017	10	84
12b73	564367	4925408	10	80
12b15	566528	4925343	11	81
10B294	564714	4918101	12	71
10B350	562181	4914854	12	70
12b9	565249	4925632	12	90
257	563383	4918160	13	80
192	563018	4911014	13	80
281	562606	4922570	13	74
12b74	564609	4925466	13	85
12b74	564609	4925466	13	84
12b33	563237	4916238	14	65
10B252	563182	4921127	15	76
10B335	562734	4910230	15	75
353	563856	4924651	15	70
158	560823	4920948	15	80
12b1	564064	4919837	15	75
12b22	564910	4922012	15	83
256	564196	4917699	16	76
12b41	561281	4913879	16	80
12b74	564609	4925466	16	81

10B278	565277	4920296	18	78
307	563652	4920458	18	75
311	563819	4921753	18	86
270	563203	4914508	18	80
265	562183	4914853	18	72
12b29	563356	4911550	18	82
10B343	562093	4917361	19	69
226	564325	4916243	19	68
343	564066	4922584	20	74
347	564202	4923510	20	85
232	563145	4918523	20	76
10b339	562916	4916258	21	77
317	561254	4915050	21	78
12b35	562083	4916519	21	73
12b42	562278	4914361	21	85
200	562424	4910135	22	78
215	561358	4911562	22	82
222	562320	4911899	22	84
12b92	559931	4914940	22	80
252	564360	4919255	23	83
267	563278	4914963	23	84
219	562149	4912372	23	76
262	561058	4915442	23	70
12b34	562545	4916573	23	80
12b37	561122	4915981	23	83
12b20	563579	4921650	24	82
12b53	559727	4915063	24	84
213	562414	4910680	25	80
217	561601	4911802	25	75
12b21	564148	4922080	25	84
291	561226	4921280	26	78
319	563260	4923531	26	88
12b31	562199	4911178	26	78
12b40	560441	4914577	26	78
12b81	561353	4911377	26	84
244	562086	4916520	27	74
183	562385	4911331	28	70
246	561112	4916706	28	84
280	562665	4920857	28	74
284	562343	4920861	28	82

	306	563613	4920114	30	79
	242	563007	4916431	30	80
	338	560845	4921119	30	76
12b32		561675	4911115	30	85
	254	564448	4918489	31	82
	261	560811	4915223	31	70
	228	564391	4916670	32	72
	230	564096	4915684	32	82
	313	564469	4921361	33	78
	205	563502	4913000	33	82
	272	562810	4914633	34	85
	314	564291	4920918	35	84
	334	564259	4925347	35	80
12b30		562269	4911430	35	84
	214	561405	4911320	36	80
	247	561082	4916561	36	76
	250	563828	4919173	38	85
	255	564141	4918083	43	80
	354	564722	4924615	45	83
12b38		560845	4916189	190	20
10B305		562397	4911536	233	25
	241	561125	4917141	240	74
10B349		562041	4915544	350	78
10B297		564330	4917548	353	74
10B321		561292	4917601	353	75
10B286		564402	4919040	354	66
10B346		561624	4916926	354	66
12b73		564367	4925408	354	79
10B312		562092	4913318	358	78
	290	561247	4921925	358	81
12b7		564258	4925370	358	85
12b6		564099	4925419	359	85

765

766 *West Mountain Foliation-lineation pairs*

location	easting	northing	strike	dip	rake
10NB251	563187	4920843	0	75	84
10NB296	564303	4917183	1	74	84
10NB288	564939	4919194	1	78	109
10NB334	562780	4910922	2	83	106

10nb15	563700	4914745	2	79	70
10NB353	563146	4916000	2	70	86
10NB259	564014	4916127	2	80	94
10NB284	564090	4918880	2	76	68
12nb58	565161	4925777	3	90	90
10NB351	562545	4914749	4	70	83
10NB295	564581	4916838	4	74	94
10NB345	561750	4917349	4	79	81
12nb74	564609	4925466	4	84	88
12nb14	565854	4925645	4	82	87
10NB318	562379	4918476	5	87	85
10NB274	563763	4919693	5	68	49
292	561008	4921378	5	68	88
12nb8	565187	4925595	5	84	90
10NB303	563145	4911642	6	74	88
10NB320	561771	4918228	6	83	106
10NB282	563709	4918710	6	74	89
10nb256	562933	4920045	6	82	76
10NB254	563759	4921942	6	76	107
10NB255	564296	4922089	6	72	81
10NB299	563764	4917245	7	66	77
316	559640	4915004	8	77	79
225	563998	4915962	8	74	83
233	562818	4919310	8	76	76
10NB270	562907	4920261	8	67	60
12nb73	564367	4925408	8	83	90
12nb9	565249	4925632	8	79	92
10NB304	562493	4911386	9	81	96
10NB283	564069	4918528	9	55	64
10NB272	562508	4920909	9	75	104
10nb21	560176	4914984	10	74	98
10NB341	562284	4916038	10	55	80
10NB322	561574	4917466	10	53	75
10NB275	564084	4919796	10	74	86
10NB276	564369	4920044	10	71	94
12nb48	561006	4921364	10	65	100
337	564778	4925527	10	81	94
237	561946	4918274	11	86	72
10NB285	564201	4919037	11	78	71
348	563819	4923745	11	85	77

10NB271	562765	4920729	13	70	94
10NB302	563437	4911950	14	67	97
10NB323	561806	4917596	14	75	86
10NB277	564575	4920338	14	69	87
10NB306	561585	4911310	15	75	101
10NB315	562219	4912076	15	77	102
10NB314	562119	4912641	15	80	87
10NB313	562060	4912935	15	76	53
10NB348	562017	4916269	15	67	68
287	560766	4919763	15	83	99
10NB336	562484	4909840	16	65	86
10NB337	562862	4910336	16	84	106
253	564162	4918708	16	82	78
10NB287	564619	4919104	16	83	92
345	564813	4922999	16	86	120
12nb6	564099	4925419	16	80	82
10NB309	561398	4912922	17	75	96
10NB300	563630	4917000	17	76	98
10NB269	562537	4920037	17	76	95
180	563219	4911445	18	78	60
206	563355	4913162	18	76	98
271	562921	4914587	18	84	72
10nb20	559716	4915012	18	80	74
227	564460	4916503	18	70	70
243	562536	4916523	18	85	103
276	562903	4920219	18	78	86
283	562139	4921509	18	80	96
341	563129	4922574	18	70	80
293	561411	4922912	18	84	72
304	561572	4923631	18	75	72
204	563447	4912616	19	77	98
10NB311	562016	4913416	19	80	86
278	562773	4920762	19	70	83
10NB253	563408	4921898	19	74	72
245	561466	4916675	20	65	96
10NB250	563281	4920358	20	74	83
336	561302	4925354	20	82	88
203	563465	4912283	21	78	102
10NB308	561391	4912787	21	77	98
181	562848	4911146	22	82	102

10nb19	562197	4911926	22	79	90
273	562499	4919385	22	84	88
275	562606	4920092	22	79	98
344	564770	4922618	22	85	88
294	561498	4923408	23	84	86
349	563386	4923719	23	83	76
224	563924	4915661	24	75	82
251	564046	4919273	24	82	73
310	563570	4921601	24	81	96
191	563032	4911027	25	80	92
207	563206	4913291	25	85	105
12nb36	561614	4916371	25	85	80
236	562544	4918749	25	82	65
335	564258	4925347	25	68	90
216	561404	4911631	27	71	102
258	560911	4916388	28	76	84
249	561643	4916771	28	71	100
309	563594	4921409	28	84	98
269	563258	4914518	29	80	83
218	561763	4912260	32	78	104
268	563536	4914451	32	82	100
202	563512	4912056	33	85	84
318	561510	4912827	36	77	73
185	562022	4911292	38	55	98
229	564252	4916791	38	80	75
10NB301	563768	4916433	166	78	116
10NB333	562945	4911224	169	85	105
10NB319	562177	4918387	171	83	87
10NB376	562668	4919460	175	79	90
10NB257	562733	4919640	181	72	80
10NB268	562491	4919388	197	88	98
10nb17	563439	4913213	330	52	102
10NB325	562225	4917845	350	86	105
10NB331	563148	4910998	352	75	79
315	560210	4915033	353	85	83
10NB327	562898	4918849	353	79	87
10nb16	562864	4915344	354	82	98
10NB310	561791	4913562	355	75	101
10NB342	562374	4916633	355	80	94
10NB316	563126	4919207	355	62	104

12nb27	565758	4922420	355	75	93
10NB326	562682	4918413	356	75	83
10NB273	563192	4921417	356	75	95
10NB344	562264	4917415	357	79	92
10NB324	561897	4917860	357	61	100

767

768 *APPENDIX 4: West Mountain statistical analysis*

769 *### PRELIMINARY WORK ###*

770

771 # Set the working directory.

772 setwd("~/Desktop/20170620geologyGeometry")

773 # Load the necessary R libraries.

774 source("library/all.R")

775

776 # Load any custom functions called in this file####

777

778 #one Regression does a regression based on an azimuth (e.g. how well
779 does northeasting (045) explain variation in my data?) This kernal
780 geodetic regression is analogous to a best-fit line, and thus describes
781 a steady variation, not a sudden change in orientation.

782 oneRegression <-

783 function(follins,

784 domain,

785 pValuePerms,

786 directionality) {

787 regression <-

788 oriGeodesicRegression(

789 cos((directionality) * pi / 180) * follins\$northing[domain] +

790 sin((directionality) *

791

792 pi / 180) * follins\$easting[domain],

793 follins\$rotation[domain],

794 oriLineInPlaneGroup,

795 numSteps = 10000

796)

797

798 if (pValuePerms > 0) {

799 RSquareds <-

800 oriGeodesicRegressionPermutations(

801 cos((directionality) * pi / 180) * follins\$northing[domain] +

802 sin((directionality) *

803

804 pi / 180) * follins\$easting[domain],

805 follins\$rotation[domain],

806 numPerms = pValuePerms,

807 group = oriLineInPlaneGroup

808)

809 sum(RSquareds > regression\$rSquared)

810 p <- sum(RSquareds > regression\$rSquared) / length(RSquareds)

811 } else {

```

812     p <- 0
813   }
814   regressionStats <- zeros(1, 5)
815   regressionStats[1, 1] = directionality
816   regressionStats[1, 2] = regression$error
817   regressionStats[1, 3] = regression$minEigenvalue
818   regressionStats[1, 4] = regression$rSquared
819   regressionStats[1, 5] = p
820   names(regressionStats) = c("azimuth", "error", "minEigenValue",
821 "R^2", "P")
822   regression$stats <- regressionStats
823   return(regression)
824 }
825
826 #regressionSweep does a series of directional regressions from 0-180°,
827 given an increment, and also calculates the pValue for each regression.
828 BEWARE: doing this with p-values can take days-weeks-or-months of
829 computing time. I suggest first running the function with degreeIncrement
830 = 10 and pValuePerms=0. This may still take a couple hours, but will
831 give you a first order picture of whether it is interesting to proceed.
832 regressionSweep <-
833   function(follins,
834           degreeIncrement,
835           domain,
836           pValuePerms) {
837     v <- rbind(1)
838     intervals <- 180 / degreeIncrement
839     for (x in 2:intervals) {
840       vTemp <- x
841       v <- rbind(v, vTemp)
842     }
843     v <- as.matrix(as.vector(v))
844     # Geodesic regression of pole vs. northing in domain 4. Check that
845     error is 0 and minEigenvalue is positive.
846     regressions <- list()
847     regressionStats <- zeros(nrow(v), 5)
848     for (i in 1:nrow(v)) {
849       regressionTemp <-
850         oriGeodesicRegression(
851           cos((v[i, 1] - 1) * degreeIncrement * pi / 180) *
852           follins$northing[domain] + sin((v[i, 1] -
853           1) * degreeIncrement * pi / 180) * follins$easting[domain],
854           follins$rotation[domain],
855           oriLineInPlaneGroup,
856           numSteps = 10000
857         )
858       regressions[[i]] <- regressionTemp
859       regressionStats[i, 1] = (i - 1) * degreeIncrement
860       regressionStats[i, 2] = regressionTemp$error
861       regressionStats[i, 3] = regressionTemp$minEigenvalue
862       regressionStats[i, 4] = regressionTemp$rSquared
863       if (pValuePerms > 0) {

```

```

865         RSquareds <-
866             oriGeodesicRegressionPermutations(
867                 cos((v[i, 1] - 1) * degreeIncrement * pi / 180) *
868                 follins$northing[domain] + sin((v[i, 1] - 1) * degreeIncrement * pi /
869                 180) * follins$easting[domain],
870                 follins$rotation[domain],
871                 numPerms = pValuePerms,
872                 group = oriLineInPlaneGroup)
873             length(RSquareds)
874             sum(RSquareds > regressionTemp$rSquared)
875             p <-
876                 sum(RSquareds > regressionTemp$rSquared) / length(RSquareds)
877             }
878             else {
879                 p <- "nan"
880             }
881         regressionStats[i, 5] = p
882     }
883     regressions[[i + 1]] <- regressionStats
884     return(regressions)
885 }
886 }
887
888 #take two bootstrapped mean clouds and the number of data points in each,
889 #and calculate a distribution of angular differences. Probably not the
890 #most statistically robust.
891 distHist <- function(bootsOne, bootsTwo, numPerms, group) {
892     distances <- list()
893     boots1 <- bootsOne$bootstraps
894     boots2 <- bootsTwo$bootstraps
895     for (i in 1:numPerms) {
896         a <- round(runif(1, 1, length(boots1)), 0)
897         b <- round(runif(1, 1, length(boots2)), 0)
898         distances[[i]] <- oriDistance(boots1[[a]], boots2[[b]], group)
899     }
900     return(distances)
901 }
902 }
903
904 #Using one two-sample bootstrapped mean difference cloud, calculate the
905 #distribution of angular differences. Probably not the most statistically
906 #robust.
907 diffDistHist <- function(bootsDiff, numPerms, group) {
908     boots <- bootsDiff$bootstraps
909     distances <-
910         lapply(boots, function(s)
911             oriDistance(s, diag(3), group) * 180 / pi)
912     return(distances)
913 }
914 END SCRIPT
915
916

```

```

917
918
919
920
921
922 ### PRELIMINARY WORK ####
923
924 # Set the working directory.
925 setwd("~/Desktop/20170620geologyGeometry")
926 # Load the necessary R libraries.
927 source("library/all.R")
928 # Markov chain Monte Carlo and Kamb contouring in equal volume plots
929 require C compiler. Skip MCMC and equal volume Kamb lines if you do not
930 wish to install C. Load the necessary library
931 source("libraryC/all.R")
932 # Load some custom functions.
933 source("JSG_statsFunctions.r")
934
935 #=====LOAD THE
936 DATA=====
937
938 # 1) Foliation only
939 Fols <- geoDataFromFile("data/Fols_WestMt.csv")
940
941 # Plot foliation locations in map view
942 plot(Fols$easting, Fols$northing, xlab = "Easting (meters)", ylab =
943 "Northing (meters)")
944
945 # Check how many measurments there are
946 nrow(Fols)
947
948 #-----
949
950 # 2) Foliation-lineation pairs
951 Follins <- geoDataFromFile("data/Follins_WestMt.csv")
952
953 # Plot foliation-lineation locations in map view
954 plot(
955     Follins$easting,
956     Follins$northing,
957     xlab = "Easting (meters)",
958     ylab = "Northing (meters)"
959 )
960
961 # Check how many measurements there are
962 nrow(Follins)
963
964
965
966 #=====PART I: DIRECTIONAL STATISTICS ON
967 FOLIATIONS=====

```

```

968 #=====DEFINE GEOGRAPHIC DOMAINS FOR FOLIATION-ONLY
969 DATA=====
970
971 #(After Braudy et al., 2017)
972
973 # Define the geographic criteria that defines the different domains
974 Fols_northCrit <- Fols$northing > 4922736
975 Fols_centerCrit <- Fols$northing < 4922736 & Fols$northing > 4919205
976 Fols_southCrit <- Fols$northing < 4919205
977
978 # Create a new column in the dataframe in which to store the domain
979 information
980 Fols$domain <- replicate(nrow(Fols), 1)
981
982 # Classify the foliation-only dataset by domain
983 Fols$domain[Fols_northCrit] <- 1
984 Fols$domain[Fols_centerCrit] <- 2
985 Fols$domain[Fols_southCrit] <- 3
986
987 # Plot the locations of the foliation-only data in map view, each domain
988 a different color.
989 plot(
990   x = Fols$easting,
991   y = Fols$northing,
992   xlab = "Easting (meters)",
993   ylab = "Northing (meters)",
994   col = hues(Fols$domain),
995   pch = 19
996 )
997
998 # Return the number of datapoints in each domain.
999 length(Fols$domain[Fols_northCrit])
1000 length(Fols$domain[Fols_centerCrit])
1001 length(Fols$domain[Fols_southCrit])
1002
1003 #=====DEFINE GEOGRAPHIC DOMAINS FOR FOLIATION-
1004 LINEATION DATA=====
1005
1006 # (After Braudy et al., 2017)
1007
1008 # Define the geographic criteria that defines the different domains
1009 Follins_northCrit <- Follins$northing > 4922736
1010 Follins_centerCrit <- Follins$northing < 4922736 & Follins$northing >
1011 4919205
1012 Follins_southCrit <- Follins$northing < 4919205
1013
1014 # Create a new column in the dataframe in which to store the domain
1015 information
1016 Follins$domain <- replicate(nrow(Follins), 1)
1017
1018 # Classify the foliation-lineation dataset by domain
1019 Follins$domain[Follins_northCrit] <- 1
1020 Follins$domain[Follins_centerCrit] <- 2

```

```

1021 Follins$domain[Follins_southCrit] <- 3
1022
1023 # Plot the locations of the foliation-lineation data in map view, each
1024 domain a different color.
1025 plot(
1026     x = Follins$easting,
1027     y = Follins$northing,
1028     xlab = "Easting (meters)",
1029     ylab = "Northing (meters)",
1030     col = hues(Follins$domain),
1031     pch = 19
1032 )
1033
1034 # Return the number of datapoints in each domain.
1035 length(Follins$domain[Follins_northCrit])
1036 length(Follins$domain[Follins_centerCrit])
1037 length(Follins$domain[Follins_southCrit])
1038
1039 #=====FOLIATION-ONLY                                STATISTICAL
1040 TREATMENT=====
1041
1042 # 1) Some parametric two-sample tests. The null hypothesis for all tests
1043 is that the two domains being tested come from the same population.
1044
1045 # Three Wellner tests (Wellner, 1979), one for each pair of domains.
1046 Each test is based on 10,000 permutations. This tests not only if the
1047 samples come from the populations with the same mean, but also with the
1048 same dispersion.
1049 lineWellnerInference(Fols$pole[Fols_northCrit],
1050 Fols$pole[Fols_southCrit], 10000)
1051 lineWellnerInference(Fols$pole[Fols_northCrit],
1052 Fols$pole[Fols_centerCrit], 10000)
1053 lineWellnerInference(Fols$pole[Fols_centerCrit],
1054 Fols$pole[Fols_southCrit], 10000)
1055
1056 # Three Watson tests that assume large sample size (Mardia and Jupp,
1057 2000), one for each pair of domains.
1058 lineLargeMultiSampleWatsonInference(list(Fols$pole[Fols_northCrit],
1059 Fols$pole[Fols_southCrit]))
1060 lineLargeMultiSampleWatsonInference(list(Fols$pole[Fols_northCrit],
1061 Fols$pole[Fols_centerCrit]))
1062 lineLargeMultiSampleWatsonInference(list(Fols$pole[Fols_centerCrit],
1063 Fols$pole[Fols_southCrit]))
1064
1065 # Three Watson tests that assume tightly concentrated datasets (Mardia
1066 and Jupp, 2000), one for each pair of domains.
1067 lineConcentratedMultiSampleWatsonInference(list(Fols$pole[Fols_northCr
1068 it], Fols$pole[Fols_southCrit]))
1069 lineConcentratedMultiSampleWatsonInference(list(Fols$pole[Fols_northCr
1070 it], Fols$pole[Fols_centerCrit]))
1071 lineConcentratedMultiSampleWatsonInference(list(Fols$pole[Fols_centerC
1072 rit], Fols$pole[Fols_southCrit]))
1073

```

```

1074 #-----
1075
1076 # 2) Non-parametric bootstrapping
1077
1078 # Perform the bootstrapping routine for each domain. Each bootstrapped
1079 dataset is based on 10,000 iterations.
1080 Fols_northBoots <- lineBootstrapInference(Fols$pole[Fols_northCrit],
1081 10000, numPoints = 50)
1082 Fols_centerBoots <- lineBootstrapInference(Fols$pole[Fols_centerCrit],
1083 10000, numPoints = 50)
1084 Fols_southBoots <- lineBootstrapInference(Fols$pole[Fols_southCrit],
1085 10000, numPoints = 50)
1086
1087 # Plot data for each domain. Northern (red), central (green), southern
1088 (blue)
1089 lineEqualAreaPlotThree(
1090     Fols$pole[Fols_northCrit],
1091     Fols$pole[Fols_centerCrit],
1092     Fols$pole[Fols_southCrit]
1093 )
1094
1095 # Plot the bootstrapped mean clouds for each domain. Northern (red),
1096 central (green), southern (blue)
1097 lineEqualAreaPlotThree(
1098     Fols_northBoots$bootstraps,
1099     Fols_centerBoots$bootstraps,
1100     Fols_southBoots$bootstraps
1101 )
1102
1103 # Plot the bootstrapped mean clouds with 95% confidence ellipses
1104 superimposed. Northern (red), central (green), southern (blue)
1105 lineEqualAreaPlotThree(
1106     list(Fols_northBoots$center),
1107     list(Fols_centerBoots$center),
1108     list(Fols_southBoots$center),
1109     curves = list(
1110         Fols_northBoots$points,
1111         Fols_centerBoots$points,
1112         Fols_southBoots$points
1113     )
1114 )
1115
1116 #-----
1117
1118 # 3) Compute the rotation between the north and south domains.
1119
1120 # Create empty lists to store data
1121 northSouthDiff <- list()
1122
1123 # Assign variables for the orientations of all bootstrapped means.
1124 northB <- Fols_northBoots$bootstraps
1125 southB <- Fols_southBoots$bootstraps
1126

```



```

1127 # Calculate the smallest possible rotations that bring means from the
1128 northern bootstrap cloud to the southern bootstrap clouds.
1129 count = 1
1130 for (i in 1:100) {
1131     for (n in i:100) {
1132         northSouthDiff$rotation[[count]] =
1133         rotSmallestRotationFromTwoLines(northB[[i]], southB[[n]])
1134         if (n < 1000) {
1135             count = count + 1
1136         }
1137     }
1138 }
1139
1140 # Display the raw results of the preceting for loop
1141 northSouthDiff
1142
1143 # Plot these rotations in an equal volume plot
1144 oriEqualVolumePlot(
1145     northSouthDiff$rotation,
1146     group = oriLineInPlaneGroup,
1147     simplePoints = TRUE
1148 )
1149
1150 # Compute the axis and rotation amount from all the rotations
1151 Fols_angleAxis <- lapply(northSouthDiff$rotation, function(s)
1152 rotAxisAngleFromMatrix(s))
1153
1154 # Plot the rotation axes in an equal area plot
1155 lineEqualAreaPlot(
1156     lapply(Fols_angleAxis, function(s) c(s[1], s[2],
1157 s[3])),
1158     shapes = '.'
1159 )
1160
1161 # Plot a histogram of rotation amount between the northern domain and
1162 the southern domain, in degrees.
1163 hist(
1164     as.numeric(lapply(Fols_angleAxis, function(s) s[4] * 180 / pi)),
1165     30,
1166     xlab = "Angular Distance (degrees)",
1167     ylab = "Frequency",
1168     main = "Angular distances, Northern to Southern"
1169 )
1170
1171 # Compute the mean and 1-sigma standard deviation of the rotation amount,
1172 in degrees
1173 mean(sapply(Fols_angleAxis, function(s) s[4] * 180 / pi))
1174 sd(sapply(Fols_angleAxis, function(s) s[4] * 180 / pi))
1175
1176 # Compute the mean trend and plunge in degrees
1177 Fols_axes <- lapply(Fols_angleAxis, function(s) c(s[1], s[2], s[3]))
1178 Fols_meanAxisDeg <-
1179 geoTrendPlungeDegFromCartesian(lower(lineProjectedMean(Fols_axes)))

```

```

1180
1181 #Print the mean axis of rotation, in trend and plunge
1182 Fols_meanAxisDeg
1183
1184 #=====FOLIATIONS FROM FOLIATION-LINEATION PAIRS
1185 STATISTICAL TREATMENT=====
1186
1187 ##### FOLIATIONS FROM FOLIATION-LINEATION PAIRS STATISTICAL TREATMENT
1188
1189 # 1) Some parametric two-sample tests.
1190
1191 # Three Wellner tests (Wellner, 1979), one for each pair of domains.
1192 Each test is based on 10,000 permutations.
1193 lineWellnerInference(Follins$pole[Follins_northCrit],
1194 Follins$pole[Follins_southCrit], 10000)
1195 lineWellnerInference(Follins$pole[Follins_northCrit],
1196 Follins$pole[Follins_centerCrit], 10000)
1197 lineWellnerInference(Follins$pole[Follins_centerCrit],
1198 Follins$pole[Follins_southCrit], 10000)
1199
1200 # Three Watson tests that assume large sample size (Mardia and Jupp,
1201 2000), one for each pair of domains.
1202 lineLargeMultiSampleWatsonInference(list(Follins$pole[Follins_northCri
1203 t], Follins$pole[Follins_southCrit]))
1204 lineLargeMultiSampleWatsonInference(list(Follins$pole[Follins_northCri
1205 t], Follins$pole[Follins_centerCrit]))
1206 lineLargeMultiSampleWatsonInference(list(Follins$pole[Follins_centerCr
1207 it], Follins$pole[Follins_southCrit]))
1208
1209 # Three Watson tests that assume tightly concentrated datasets (Mardia
1210 and Jupp, 2000), one for each pair of domains.
1211 lineConcentratedMultiSampleWatsonInference(list(Follins$pole[Follins_n
1212 orthCrit], Follins$pole[Follins_southCrit]))
1213 lineConcentratedMultiSampleWatsonInference(list(Follins$pole[Follins_n
1214 orthCrit], Follins$pole[Follins_centerCrit]))
1215 lineConcentratedMultiSampleWatsonInference(list(Follins$pole[Follins_c
1216 enterCrit], Follins$pole[Follins_southCrit]))
1217
1218 #-----
1219
1220 # 2) Non-parametric bootstrapping
1221
1222 # Perform the bootstrapping routine for each domain. Each bootstrapped
1223 dataset is based on 10,000 iterations.
1224 FollinFols_northBoots <-
1225 lineBootstrapInference(Follins$pole[Follins_northCrit], 10000,
1226 numPoints = 50)
1227 FollinFols_centerBoots <-
1228 lineBootstrapInference(Follins$pole[Follins_centerCrit], 10000,
1229 numPoints = 50)
1230 FollinFols_southBoots <-
1231 lineBootstrapInference(Follins$pole[Follins_southCrit], 10000,
1232 numPoints = 50)

```

```

1233
1234
1235 # Plot data for each domain. Northern (red), central (green), southern
1236 (blue)
1237 lineEqualAreaPlotThree(
1238     Follins$pole[Follins_northCrit],
1239     Follins$pole[Follins_centerCrit],
1240     Follins$pole[Follins_southCrit]
1241 )
1242
1243 # Plot the bootstrapped mean clouds for each domain. Northern (red),
1244 central (green), southern (blue)
1245 lineEqualAreaPlotThree(
1246     FollinFols_northBoots$bootstraps,
1247     FollinFols_centerBoots$bootstraps,
1248     FollinFols_southBoots$bootstraps
1249 )
1250
1251 # Plot the bootstrapped mean clouds with 95% confidence ellipses
1252 superimposed. Northern (red), central (green), southern (blue)
1253 lineEqualAreaPlotThree(
1254     list(FollinFols_northBoots$center),
1255     list(FollinFols_centerBoots$center),
1256     list(FollinFols_southBoots$center),
1257     curves = list(
1258         FollinFols_northBoots$points,
1259         FollinFols_centerBoots$points,
1260         FollinFols_southBoots$points
1261     )
1262 )
1263
1264 # -----
1265
1266 # 3) Compute the rotation between the north and south domains.
1267
1268 # Create empty lists to store data
1269 Follins_northSouthDif <- list()
1270
1271 # Assign variables for the orientations of all bootstrapped means.
1272 Follins_northB <- FollinFols_northBoots$bootstraps
1273 Follins_southB <- FollinFols_southBoots$bootstraps
1274
1275 # Calculate the smallest possible rotations that bring means from the
1276 northern bootstrap cloud to the southern bootstrap clouds.
1277 count = 1
1278 for (i in 1:100) {
1279     for (n in i:100) {
1280         Follins_northSouthDif$rotation[[count]] =
1281         rotSmallestRotationFromTwoLines(Follins_northB[[i]],
1282         Follins_southB[[n]])
1283         if (n < 1000) {
1284             count = count + 1
1285         }
1286     }
1287 }

```

```

1286     }
1287 }
1288
1289 # Display the raw results of the preceting for loop
1290 Follins_northSouthDif
1291
1292 # Plot these rotations in an equal volume plot
1293 oriEqualVolumePlot(
1294     Follins_northSouthDif$rotation,
1295     group = oriLineInPlaneGroup,
1296     simplePoints = TRUE
1297 )
1298
1299 # Compute the axis and rotation amount from all the rotations
1300 Follins_angleAxis <- lapply(Follins_northSouthDif$rotation, function(s)
1301     rotAxisAngleFromMatrix(s))
1302
1303 # Plot the rotation axes in an equal area plot
1304 lineEqualAreaPlot(lapply(Follins_angleAxis, function(s) c(s[1], s[2],
1305     s[3])), shapes = '.')
1306
1307 # Plot a histogram of rotation amount between the northern domain and
1308 the southern domain, in degrees.
1309 hist(
1310     as.numeric(lapply(Follins_angleAxis, function(s)
1311         s[4] * 180 / pi)),
1312     30,
1313     xlab = "Angular Distance (degrees)",
1314     ylab = "Frequency",
1315     main = "Angular distances, Northern to Southern"
1316 )
1317
1318 # Compute the mean and 1-sigma standard deviation of the rotation amount,
1319 in Follins_angleAxisrees
1320 mean(sapply(Follins_angleAxis, function(s) s[4] * 180 / pi))
1321 sd(sapply(Follins_angleAxis, function(s) s[4] * 180 / pi))
1322
1323 # Compute the mean trend and plunge in Follins_angleAxisrees
1324 Follins_axes <- lapply(Follins_angleAxis, function(s) c(s[1], s[2],
1325     s[3]))
1326 Follins_meanAxisDeg <-
1327     geoTrendPlungeDegFromCartesian(lower(lineProjectedMean(Follins_axes)))
1328
1329 #Print the mean axis of rotation, in trend and plunge
1330 Follins_meanAxisDeg
1331
1332 #=====COMPARE FOLIATIONS FROM THE TWO
1333 DATASETS=====
1334
1335 # (Foliation-only vs. Foliation-lineation datasets)
1336

```

```

1337 # 1) Three plots to compare the foliations of the Fols (Foliation only
1338 data set) and Follins (Foliation-lineation dataset) in the northern
1339 domain
1340
1341 # Equal area plot of the Fols (cyan) and Follins (red)
1342 lineEqualAreaPlotTwo(Follins$pole[Follins_northCrit],
1343 Fols$pole[Fols_northCrit])
1344
1345 # Equal area plot of the bootstrapped means for the Fols (cyan) and
1346 Follins (red)
1347 lineEqualAreaPlotTwo(
1348     FollinFols_northBoots$bootstraps,
1349     Fols_northBoots$bootstraps
1350 )
1351
1352 # Equal area plot of 95% confidence ellipses from bootstrapping for the
1353 Fols (cyan) and Follins (red)
1354 lineEqualAreaPlotTwo(
1355     list(FollinFols_northBoots$center),
1356     list(Fols_northBoots$center),
1357     curves = list(FollinFols_northBoots$points,
1358                   Fols_northBoots$points
1359 )
1360 )
1361
1362 #-----
1363
1364 # 2) Three plots to compare the foliations of the Fols (Foliation only
1365 data set) and Follins (Foliation-lineation dataset) in the central domain
1366
1367 # Equal area plot of the Fols (cyan) and Follins (red)
1368 lineEqualAreaPlotTwo(
1369     Follins$pole[Follins_centerCrit],
1370     Fols$pole[Fols_centerCrit]
1371 )
1372
1373 # Equal area plot of the bootstrapped means for the Fols (cyan) and
1374 Follins (red)
1375 lineEqualAreaPlotTwo(
1376     FollinFols_centerBoots$bootstraps,
1377     Fols_centerBoots$bootstraps
1378 )
1379
1380 # Equal area plot of 95% confidence ellipses from bootstrapping for the
1381 Fols (cyan) and Follins (red)
1382 lineEqualAreaPlotTwo(
1383     list(FollinFols_centerBoots$center),
1384     list(Fols_centerBoots$center),
1385     curves = list(FollinFols_centerBoots$points,
1386                   Fols_centerBoots$points
1387 )
1388 )
1389

```

```

1390 #-----
1391
1392 # 3) Three plots to compare the foliations of the Fols (Foliation only
1393 data set) and Follins (Foliation-lineation dataset) in the southern
1394 domain
1395
1396 # Equal area plot of the Fols (cyan) and Follins (red)
1397 lineEqualAreaPlotTwo(
1398     Follins$pole[Follins_southCrit],
1399     Fols$pole[Fols_southCrit]
1400 )
1401
1402 # Equal area plot of the bootstrapped means for the Fols (cyan) and
1403 Follins (red)
1404 lineEqualAreaPlotTwo(
1405     FollinFols_southBoots$bootstraps,
1406     Fols_southBoots$bootstraps
1407 )
1408
1409 # Equal area plot of 95% confidence ellipses from bootstrapping for the
1410 Fols (cyan) and Follins (red)
1411 lineEqualAreaPlotTwo(
1412     list(FollinFols_southBoots$center),
1413     list(Fols_southBoots$center),
1414     curves = list(FollinFols_southBoots$points,
1415                   Fols_southBoots$points
1416     )
1417 )
1418
1419
1420 #=====
1421 #=====
1422
1423
1424
1425 #=====PART II: ORIENTATION STATISTICS ON FOLIATION-
1426 LINEATION PAIRS=====
1427
1428 #These are all foliation-lineation pairs, so they are orientation data.
1429 We will proceed with methods outlined in Davis and Titus, 2017.
1430
1431 #Since lines on foliation are bidirectional, these are line-in-plane
1432 data, and have a four fold symmetry (see Davis and Titus, 2017).
1433
1434 #=====PLOT THE
1435 DATA=====
1436
1437 # Plot the data in Equal area plot. Lineation (red), Pole to foliation
1438 (cyan)
1439 lineEqualAreaPlotTwo(
1440     Follins$direction,
1441     Follins$pole
1442 )

```

```

1443
1444 # Plot the data in an Equal Volume Plot (after Davis and Titus, 2017).
1445 Each point represents a foliation-lineation pair. (There are four copies
1446 of the data due to mathematical symmetry)
1447 oriEqualVolumePlot(
1448     Follins$rotation,
1449     oriLineInPlaneGroup
1450 )
1451
1452 # We can also do some basic directional kamb contouring for poles and
1453 lines. The numbers are the kamb intervals
1454 lineKambPlot(
1455     Follins$pole,
1456     c(2, 4, 6, 8, 10, 12, 14, 18)
1457 )
1458 lineKambPlot(
1459     Follins$direction,
1460     c(2, 4, 6, 8, 10, 12, 14, 18)
1461 )
1462
1463 # NOTE: This line requires c library to run--skip if you have not compiled
1464 c.
1465 # Uncomment lines below to Plot 6-sigma kamb contour in an Equal volume
1466 plot of Follins.
1467
1468 # oricKambPlot(Follins$rotation,
1469 #             group=oriLineInPlaneGroup,
1470 #             multiple = 6,
1471 #             simplePoints = TRUE,
1472 #             backgroundColor="white", curveColor="black",
1473 #             boundaryAlpha=0,
1474 #             colors="black", axesColors=c("black", "black", "black"),
1475 #             fogStyle="exp")
1476
1477 # If you wish to save this figure, maximize the plot window on your
1478 screen before running this line. It will save to your working directory
1479 folder.
1480
1481 #             afterMaximizingWindow("WestMt_kamb_EqualVol1.png",
1482 "WestMt_kamb_EqualVol2.png")
1483
1484
1485 #=====IDENTIFY                                GEOGRAPHIC
1486 TRENDS=====
1487
1488 # 1) Visually inspect the possibility of geographic trends in the data
1489
1490 # Plot an Equal Volume plot (after Davis and Titus, 2017), colored by
1491 the three domains. Northern (red); Central (green); Southern (blue).
1492 oriEqualVolumePlot(
1493     Follins$rotation,
1494     group = oriLineInPlaneGroup,
1495     backgroundColor = "white",

```

```

1496         curveColor = "black",
1497         boundaryAlpha = 0.2,
1498         colors = hues(Follins$domain),
1499         axesColors = c("black", "black", "black")
1500     )
1501
1502 # Plot an Equal volume plot of the data, colored by northing
1503 oriEqualVolumePlot(
1504     Follins$rotation,
1505     group = oriLineInPlaneGroup,
1506     backgroundColor = "white",
1507     curveColor = "black",
1508     boundaryAlpha = 0.2,
1509     colors = hues(Follins$northing),
1510     axesColors = c("black", "black", "black")
1511 )
1512
1513 # Plot an Equal volume plot of the data, colored by easting
1514 oriEqualVolumePlot(
1515     Follins$rotation,
1516     group = oriLineInPlaneGroup,
1517     backgroundColor = "white",
1518     curveColor = "black",
1519     boundaryAlpha = 0.2,
1520     colors = hues(Follins$easting),
1521     axesColors = c("black", "black", "black")
1522 )
1523
1524 # Save EqualVolumePlot figure. If you wish to save this figure, maximize
1525 the plot window on your screen before running this line. It will save
1526 to your working directory folder.
1527 afterMaximizingWindow(
1528     "WestMt_domains_EqualVol.png",
1529     "WestMt_domains_EqualVol.png"
1530 )
1531
1532 # Plot Follins in an Equal area plot, colored by domain. Lineation
1533 (squares); Foliation (circles)
1534 lineEqualAreaPlot(
1535     c(Follins$pole, Follins$direction),
1536     col = hues(Follins$domain),
1537     shapes = c(replicate(length(Follins$pole), "c"),
1538               replicate(length(Follins$direction), "s"))
1539 )
1540
1541 # _____
1542
1543 # 2) Run regressions to quantify any extant trends.
1544
1545 # 2A) Geodesic regression on all data. Perform a geodesic regression
1546 with respect to azimuths every 10 degrees. Each regression asks the
1547 question: "does orientation change linearly with respect to an azimuth
1548 towards x degrees"

```



```

1549 regressionSWestAll <- regressionSweep(Follins, 10, Follins_northCrit |
1550 Follins_centerCrit | Follins_southCrit , 0)
1551 westAllRegSum <- data.frame(regressionSWestAll[[19]])
1552 names(westAllRegSum) <- c("Azimuth", "Error", "Min_Eigen", "R_squared",
1553 "Pvalue")
1554
1555
1556 # Plot the R^2 value (y-axis) as a function of the azimuth in degrees
1557 (x-axis)
1558 plot(
1559     as.vector(westAllRegSum[, 1]),
1560     as.vector(westAllRegSum[, 4]),
1561     main = "Geodesic regressions (all Domains)",
1562     xlab = "Azimuth in degrees",
1563     ylab = "R-squared"
1564 )
1565
1566 #-----
1567
1568 #2B) Perform geodesic regressions on each domain
1569
1570 # Perform a geodesic regression for the northern domain with respect to
1571 azimuths every 10 degrees.
1572 regressionSNorth <- regressionSweep(Follins, 10, Follins_northCrit, 0)
1573 NorthRegSum <- data.frame(regressionSNorth[[19]])
1574 names(NorthRegSum) = c("Azimuth", "Error", "Min_Eigen", "R_squared",
1575 "Pvalue")
1576
1577 # Check that "Error" = 0, "Min_Eigen" always > 0.
1578 NorthRegSum
1579
1580 # Plot the R^2 value (y-axis) as a function of the azimuth in degrees
1581 (x-axis)
1582 plot(
1583     as.vector(NorthRegSum$Azimuth),
1584     as.vector(NorthRegSum$R_squared),
1585     main = "Geodesic regressions (Northern Domain)",
1586     xlab = "Azimuth in degrees",
1587     ylab = "R-squared"
1588 )
1589
1590 #-----
1591
1592 # Perform a geodesic regression for the southern domain with respect to
1593 azimuths every 10 degrees.
1594 regressionSCenter <- regressionSweep(Follins, 10, Follins_centerCrit, 0)
1595 CenterRegSum <- data.frame(regressionSCenter[[19]])
1596 names(CenterRegSum) = c("Azimuth", "Error", "Min_Eigen", "R_squared",
1597 "Pvalue")
1598
1599 # Check that "Error" = 0, "Min_Eigen" always > 0.
1600 CenterRegSum
1601

```

```

1602 # Plot the R^2 value (y-axis) as a function of the azimuth in degrees
1603 (x-axis)
1604 plot(
1605     as.vector(CenterRegSum$Azimuth),
1606     as.vector(CenterRegSum$R_squared),
1607     main = "Geodesic regressions (Central Domain)",
1608     xlab = "Azimuth in degrees",
1609     ylab = "R-squared"
1610 )
1611
1612 #-----
1613
1614 # Perform a geodesic regression for the southern domain with respect to
1615 azimuths every 10 degrees.
1616 regressionSSouth <- regressionSweep(Follins, 10, Follins_southCrit, 0)
1617 SouthRegSum <- data.frame(regressionSSouth[[19]])
1618 names(SouthRegSum) = c("Azimuth", "Error", "Min_Eigen", "R_squared",
1619 "Pvalue")
1620
1621 # Check that "Error" = 0, "Min_Eigen" always > 0.
1622 SouthRegSum
1623
1624 # Plot the R^2 value (y-axis) as a function of the azimuth in degrees
1625 (x-axis)
1626 plot(
1627     as.vector(SouthRegSum$Azimuth),
1628     as.vector(SouthRegSum$R_squared),
1629     main = "Geodesic regressions (Southern Domain)",
1630     xlab = "Azimuth in degrees",
1631     ylab = "R-squared"
1632 )
1633
1634 #-----
1635
1636 # 3) Perform a Kernal regression for all Follins.
1637
1638 # Recategorize the symmetric copies of the data.
1639 mu <- oriFrechetMean(Follins$rotation, group=oriLineInPlaneGroup)
1640 Follins$rotation <- oriNearestRepresentatives(Follins$rotation, mu,
1641 group=oriLineInPlaneGroup)
1642
1643 # A precomputed value for bandwidth--use this value to save time.
1644
1645 bandwidth <- 0.5879134
1646
1647 # Uncomment line below to compute the appropriate bandwidth for Kernal
1648 regression with respect to easting.
1649 bandwidth <-
1650 rotBandwidthForKernelRegression(Follins$easting, Follins$rotation,
1651 dnorm)
1652
1653 #Run the kernel regression to generate a bunch of points. There were no
1654 errors and all minEigenvalues were >0

```

```

1655 kernelReg <- lapply(
1656     seq(from = min(Follins$easting), to =
1657 max(Follins$easting), by = 100),
1658     rotKernelRegression,
1659     Follins$easting,
1660     Follins$rotation,
1661     bandwidth,
1662     numSteps = 1000
1663 )
1664 sapply(kernelReg, function(regr) regr$error)
1665 sapply(kernelReg, function(regr) {regr$minEigenvalue > 0})
1666
1667 # Plot the regression curve.
1668 kernelRegCurve <- lapply(kernelReg, function(regr) regr$r)
1669 oriEqualVolumePlot(
1670     kernelRegCurve,
1671     simplePoints = TRUE,
1672     oriLineInPlaneGroup
1673 )
1674
1675 # Compute the R^2.
1676 KernRsquared <- rotRsquaredForKernelRegression(Follins$easting,
1677 Follins$rotation, bandwidth, numSteps =
1678     1000)
1679 KernRsquared
1680
1681 # Do a permutation test for significance. For the sake of time, we do
1682 only 10 permutations, although that is far too few to tell us anything.
1683 rSquareds <- rotKernelRegressionPermutations(Follins$easting,
1684 Follins$rotation, bandwidth, numPerms =
1685     1000)
1686 length(rSquareds)
1687 p <- sum(rSquareds > KernRsquared$rSquared)
1688 p
1689
1690
1691 #=====STATISTICAL
1692 DESCRIPTORS=====
1693
1694 # 1) Northern Domain, n = 16
1695
1696 # Frechet mean (minimizes the Frechet variance).
1697 FrechetMeanNorth <- oriFrechetMean(Follins$rotation[Follins_northCrit],
1698 oriLineInPlaneGroup)
1699 FrechetVarNorth <- oriVariance(Follins$rotation[Follins_northCrit],
1700 FrechetMeanNorth, oriLineInPlaneGroup)
1701
1702 # Plot the FrechetMean in an Equal Volume plot
1703 FrechetCurvesNorth <- lapply(Follins$rotation[Follins_northCrit],
1704 function(r) rotGeodesicPoints(FrechetMeanNorth, r, 10))
1705 rotEqualAnglePlot(points = Follins$rotation[Follins_northCrit], curves
1706 = FrechetCurvesNorth)
1707

```

```

1708 # Print the Strike, Dip, Rake of the Frechet mean.
1709 geoStrikeDipRakeDegFromRotation(FrechetMeanNorth)
1710
1711 # Print the Frechet variance.
1712 FrechetVarNorth
1713
1714 #-----
1715
1716 # 2) Central Domain, n = 34
1717 #Frechet mean minimizes the Frechet variance.
1718 FrechetMeanCenter <-
1719 oriFrechetMean(Follins$rotation[Follins_centerCrit],
1720 oriLineInPlaneGroup)
1721 FrechetVarCenter <- oriVariance(Follins$rotation[Follins_centerCrit],
1722 FrechetMeanCenter, oriLineInPlaneGroup)
1723
1724 #plot the FrechetMean
1725 FrechetCurvesCenter <- lapply(Follins$rotation[Follins_centerCrit],
1726 function(r) rotGeodesicPoints(FrechetMeanCenter, r, 10))
1727 rotEqualAnglePlot(
1728     points = Follins$rotation[Follins_centerCrit],
1729     curves = FrechetCurvesCenter
1730 )
1731
1732 # Print the Strike, Dip, Rake of the Frechet mean.
1733 geoStrikeDipRakeDegFromRotation(FrechetMeanCenter)
1734
1735 # Print the Frechet variance.
1736 FrechetVarCenter
1737
1738 #-----
1739
1740 # 3) Southern Domain, n = 79
1741 #Frechet mean minimizes the Frechet variance.
1742 FrechetMeanSouth <- oriFrechetMean(Follins$rotation[Follins_southCrit],
1743 oriLineInPlaneGroup)
1744 FrechetVarSouth <- oriVariance(Follins$rotation[Follins_southCrit],
1745 FrechetMeanSouth, oriLineInPlaneGroup)
1746
1747 #plot the FrechetMean
1748 FrechetCurvesSouth <- lapply(Follins$rotation[Follins_southCrit],
1749 function(r) rotGeodesicPoints(FrechetMeanSouth, r, 10))
1750 rotEqualAnglePlot(points = Follins$rotation[Follins_southCrit], curves
1751 = FrechetCurvesSouth)
1752
1753 # Print the Strike, Dip, Rake of the Frechet mean.
1754 geoStrikeDipRakeDegFromRotation(FrechetMeanSouth)
1755
1756 # Print the Frechet variance.
1757 FrechetVarSouth
1758
1759 #-----
1760

```

```

1761 # 4) Dispersion of the data
1762
1763 #The standard way to get at dispersion is to compute the maximum matrix
1764 fisher likelihood. The matrix fisher distribution comprises a kind of
1765 "mean" and a positive definite symmetric matrix, which characterises the
1766 anisotropic dispersion.
1767
1768 # Northern Domain, n = 16. Fisher maximum likelihood
1769 mleNorth <- rotFisherMLE(Follins$rotation[Follins_northCrit])
1770 mleNorth
1771 eigen(mleNorth$kHat, symmetric = TRUE, only.value = TRUE)$values
1772
1773 #Central Domain, n = 34. Fisher maximum likelihood
1774 mleCenter <- rotFisherMLE(Follins$rotation[Follins_centerCrit])
1775 mleCenter
1776 eigen(mleCenter$kHat, symmetric = TRUE, only.value = TRUE)$values
1777
1778 #Southern Domain, n = 79. Fisher maximum likelihood
1779 mleSouth <- rotFisherMLE(Follins$rotation[Follins_southCrit])
1780 mleSouth
1781 eigen(mleSouth$kHat, symmetric = TRUE, only.value = TRUE)$values
1782
1783
1784
1785 #=====INFERENCE + HYPOTHESIS
1786 TESTING=====
1787
1788 # Here, we can perform some statistical techniques to use information
1789 about the samples (each domain) to compute a probability cloud of the
1790 mean of the population(s) from which those samples were taken.
1791 # From numerical work in Davis and Titus (2017), MCMC will work the best
1792 for small sample sizes that have the matrix fisher anisotropy
1793 (eigenvalues) of (large, large, small). All four domains have this
1794 anisotropy, and have too few data points to use the (computationally
1795 quicker) bootstrapping method.
1796
1797 #We'll do both and compare.
1798
1799 # Compute the bootstrap cloud of means
1800 Follins_northBoots <-
1801 oriBootstrapInference(Follins$rotation[Follins_northCrit], 10000,
1802 oriLineInPlaneGroup)
1803 Follins_centerBoots <-
1804 oriBootstrapInference(Follins$rotation[Follins_centerCrit], 10000,
1805 oriLineInPlaneGroup)
1806 Follins_southBoots <-
1807 oriBootstrapInference(Follins$rotation[Follins_southCrit], 10000,
1808 oriLineInPlaneGroup)
1809
1810 # Compute the Markov chain Monte Carlo cloud of means
1811 northMCMC <-
1812 oricWrappedTrivariateNormalMCMCInference(Follins$rotation[Follins_northCrit],
1813

```

```

1814                                     group          =
1815 oriLineInPlaneGroup,
1816                                     numCollection   =
1817 100
1818     )
1819 centerMCMC                                                                    <-
1820 oricWrappedTrivariateNormalMCMCInference(Follins$rotation[Follins_cent
1821 erCrit],
1822                                     group          =
1823 oriLineInPlaneGroup,
1824                                     numCollection   =
1825 100
1826     )
1827 southMCMC                                                                    <-
1828 oricWrappedTrivariateNormalMCMCInference(Follins$rotation[Follins_sout
1829 hCrit],
1830                                     group          =
1831 oriLineInPlaneGroup,
1832                                     numCollection   =
1833 100
1834     )
1835
1836 #-----
1837
1838 # 1) Northern v. Central domains
1839
1840 # A) Using MCMC
1841
1842 # Construct the 95% confidence ellipsoids from small triangles
1843 trisNorthMCMC <- rotEllipsoidTriangles(northMCMC$mBar,
1844                                     northMCMC$leftCovarInv,
1845                                     northMCMC$q095,
1846                                     numNonAdapt = 4)
1847 trisCenterMCMC <- rotEllipsoidTriangles(centerMCMC$mBar,
1848                                     centerMCMC$leftCovarInv,
1849                                     centerMCMC$q095,
1850                                     numNonAdapt = 4)
1851
1852 # Plot the MCMC comparison in an equal Volume plot, with 95% confidence
1853 ellipsoids
1854 oriEqualAnglePlot(
1855     points = c(northMCMC$ms, centerMCMC$ms),
1856     boundaryAlpha = .1,
1857     axesColors = c("black", "black", "black"),
1858     fogStyle = "none",
1859     background = "white",
1860     triangles = c(trisNorthMCMC, trisCenterMCMC),
1861     simplePoints = TRUE,
1862     colors = c(replicate(length(northMCMC$ms), "black"),
1863 replicate(length(centerMCMC$ms), "orange")),
1864     group = oriTrivialGroup
1865 )
1866

```

```

1867 # If you wish to save this figure, maximize the plot window on your
1868 screen before running this line. It will save to your working directory
1869 folder.
1870 afterMaximizingWindow("MCMC_WestMt_NC_1.png", "MCMC_WestMt_NC_2.png")
1871
1872 # Plot the MCMC comparison in an Equal Area plot
1873 lineEqualAreaPlotTwo(c(
1874     lapply(northMCMC$ms, function(s)
1875         s[1,]),
1876     lapply(northMCMC$ms, function(s)
1877         s[2,])
1878 ),
1879 c(
1880     lapply(centerMCMC$ms, function(s)
1881         s[1,]),
1882     lapply(centerMCMC$ms, function(s)
1883         s[2,])
1884 ))
1885
1886 #.....
1887
1888 # B) Using bootstrapping
1889
1890 # Construct the 95% confidence ellipsoids from small triangles
1891 trisNorthBoot <-
1892     rotEllipsoidTriangles(
1893         Follins_northBoots$center,
1894         Follins_northBoots$leftCovarInv,
1895         Follins_northBoots$q095,
1896         numNonAdapt = 4
1897     )
1898 trisCenterBoot <-
1899     rotEllipsoidTriangles(
1900         Follins_centerBoots$center,
1901         Follins_centerBoots$leftCovarInv,
1902         Follins_centerBoots$q095,
1903         numNonAdapt = 4
1904     )
1905
1906 # Plot the bootstrap comparison in an equal Volume plot, with 95%
1907 confidence ellipsoids
1908 rotEqualAnglePlot(
1909     points = c(
1910         Follins_northBoots$bootstraps,
1911         Follins_centerBoots$bootstraps
1912     ),
1913     triangles = c(trisNorthBoot, trisCenterBoot),
1914     boundaryAlpha = .1,
1915     axesColors = c("black", "black", "black"),
1916     fogStyle = "none",
1917     background = "white",
1918     simplePoints = TRUE,
1919     colors = c(replicate(

```

```

1920         length(Follins_northBoots$bootstraps), "black"
1921     ), replicate(
1922         length(Follins_centerBoots$bootstraps), "orange"
1923     ))
1924 )
1925
1926 # If you wish to save this figure, maximize the plot window on your
1927 screen before running this line. It will save to your working directory
1928 folder.
1929 afterMaximizingWindow("Boots_WestMt_NC_1.png", "Boots_WestMt_NC_2.png")
1930
1931 # Plot the bootstrap comparison in an Equal Area plot
1932 lineEqualAreaPlotTwo(c(
1933     lapply(Follins_northBoots$bootstraps, function(s)
1934         s[1,]),
1935     lapply(Follins_northBoots$bootstraps, function(s)
1936         s[2,])
1937 ),
1938 c(
1939     lapply(Follins_centerBoots$bootstraps, function(s)
1940         s[1,]),
1941     lapply(Follins_centerBoots$bootstraps, function(s)
1942         s[2,])
1943 ))
1944
1945 #-----
1946
1947 # 2) Northern vs. Southern domains
1948
1949 # A) Using MCMC
1950
1951 # Construct the 95% confidence ellipsoids from small triangles
1952 trisNorthMCMC <-
1953     rotEllipsoidTriangles(northMCMC$mBar,
1954                           northMCMC$leftCovarInv,
1955                           northMCMC$q095,
1956                           numNonAdapt = 5)
1957 trisSouthMCMC <-
1958     rotEllipsoidTriangles(southMCMC$mBar,
1959                           southMCMC$leftCovarInv,
1960                           southMCMC$q095,
1961                           numNonAdapt = 5)
1962
1963 # Plot the bootstrap comparison in an equal Volume plot, with 95%
1964 confidence ellipsoids
1965 oriEqualAnglePlot(
1966     points = c(northMCMC$ms, southMCMC$ms),
1967     triangles = c(trisNorthMCMC, trisSouthMCMC),
1968     boundaryAlpha = 0.1,
1969     axesColors = c("black", "black", "black"),
1970     fogStyle = "none",
1971     background = "white",
1972     simplePoints = TRUE,

```



```

1973     colors      =      c(replicate(length(northMCMC$ms),      "black"),
1974 replicate(length(southMCMC$ms), "blue")),
1975     group = oriTrivialGroup
1976 )
1977
1978 # If you wish to save this figure, maximize the plot window on your
1979 screen before running this line. It will save to your working directory
1980 folder.
1981 afterMaximizingWindow("MCMC_WestMt_NS_1.png", "MCMC_WestMt_NS_2.png")
1982
1983 # Plot the MCMC comparison in an Equal Area plot
1984 lineEqualAreaPlotTwo(c(
1985     lapply(northMCMC$ms, function(s)
1986         s[1,]),
1987     lapply(northMCMC$ms, function(s)
1988         s[2,])
1989 ),
1990 c(
1991     lapply(southMCMC$ms, function(s)
1992         s[1,]),
1993     lapply(southMCMC$ms, function(s)
1994         s[2,])
1995 ))
1996
1997 #.....
1998
1999 # B) Using bootstrapping
2000 # Construct the 95% confidence ellipsoids from small triangles
2001 trisNorthBoot <-
2002     rotEllipsoidTriangles(
2003         Follins_northBoots$center,
2004         Follins_northBoots$leftCovarInv,
2005         Follins_northBoots$q095,
2006         numNonAdapt = 4
2007     )
2008 trisSouthBoot <-
2009     rotEllipsoidTriangles(
2010         Follins_southBoots$center,
2011         Follins_southBoots$leftCovarInv,
2012         Follins_southBoots$q095,
2013         numNonAdapt = 4
2014     )
2015
2016 # Plot the bootstrap comparison in an equal Volume plot, with 95%
2017 confidence ellipsoids.
2018 rotEqualAnglePlot(
2019     points = c(
2020         Follins_northBoots$bootstraps,
2021         Follins_southBoots$bootstraps
2022     ),
2023     triangles = c(trisNorthBoot, trisSouthBoot),
2024     boundaryAlpha = .1,
2025     axesColors = c("black", "black", "black"),

```

```

2026     fogStyle = "none",
2027     background = "white",
2028     simplePoints = TRUE,
2029     colors = c(replicate(
2030         length(Follins_northBoots$bootstraps), "black"
2031     ),
2032     replicate(
2033         length(Follins_southBoots$bootstraps), "blue"
2034     ))
2035 )
2036
2037 # If you wish to save this figure, maximize the plot window on your
2038 # screen before running this line. It will save to your working directory
2039 # folder.
2040 afterMaximizingWindow("Boots_WestMt_NS_1.png", "Boots_WestMt_NS_2.png")
2041
2042 # Plot the bootstrap comparison in an Equal Area plot
2043 lineEqualAreaPlotTwo(c(
2044     lapply(Follins_northBoots$bootstraps, function(s)
2045         s[1,]),
2046     lapply(Follins_northBoots$bootstraps, function(s)
2047         s[2,])
2048 ),
2049 c(
2050     lapply(Follins_southBoots$bootstraps, function(s)
2051         s[1,]),
2052     lapply(Follins_southBoots$bootstraps, function(s)
2053         s[2,])
2054 ))
2055
2056
2057
2058 #-----
2059
2060 # 3) Central vs. Southern domains
2061
2062 # A) Using MCMC
2063
2064 # Construct the 95% confidence ellipsoids from small triangles
2065 trisCenterMCMC <-
2066     rotEllipsoidTriangles(centerMCMC$mBar,
2067         centerMCMC$leftCovarInv,
2068         centerMCMC$q095,
2069         numNonAdapt = 4)
2070 trisSouthMCMC <-
2071     rotEllipsoidTriangles(southMCMC$mBar,
2072         southMCMC$leftCovarInv,
2073         southMCMC$q095,
2074         numNonAdapt = 4)
2075
2076 # Plot the bootstrap comparison in an equal Volume plot, with 95%
2077 # confidence ellipsoids
2078 oriEqualAnglePlot(

```

```

2079     points = c(centerMCMC$ms, southMCMC$ms),
2080     triangles = c(trisCenterMCMC, trisSouthMCMC),
2081     boundaryAlpha = 0.1,
2082     axesColors = c("black", "black", "black"),
2083     fogStyle = "none",
2084     background = "white",
2085     simplePoints = TRUE,
2086     colors      =      c(replicate(length(centerMCMC$ms),      "orange"),
2087 replicate(length(southMCMC$ms), "blue")),
2088     group = oriTrivialGroup
2089 )
2090
2091 # If you wish to save this figure, maximize the plot window on your
2092 screen before running this line. It will save to your working directory
2093 folder.
2094 afterMaximizingWindow("MCMC_WestMt_CS_1.png", "MCMC_WestMt_CS_2.png")
2095
2096 # Plot the MCMC comparison in an Equal Area plot
2097 lineEqualAreaPlotTwo(c(
2098     lapply(centerMCMC$ms, function(s)
2099         s[1,]),
2100     lapply(centerMCMC$ms, function(s)
2101         s[2,])
2102 ),
2103 c(
2104     lapply(southMCMC$ms, function(s)
2105         s[1,]),
2106     lapply(southMCMC$ms, function(s)
2107         s[2,])
2108 ))
2109
2110 #.....
2111
2112
2113 # B) Using bootstrapping
2114
2115 # Construct the 95% confidence ellipsoids from small triangles
2116 trisCenterBoot <-
2117     rotEllipsoidTriangles(
2118         Follins_centerBoots$center,
2119         Follins_centerBoots$leftCovarInv,
2120         Follins_centerBoots$q095,
2121         numNonAdapt = 4
2122     )
2123 trisSouthBoot <-
2124     rotEllipsoidTriangles(
2125         Follins_southBoots$center,
2126         Follins_southBoots$leftCovarInv,
2127         Follins_southBoots$q095,
2128         numNonAdapt = 4
2129     )
2130

```

```

2131 # Plot the bootstrap comparison in an equal Volume plot, with 95%
2132 confidence ellipsoids
2133 oriEqualAnglePlot(
2134     points = c(
2135         Follins_centerBoots$bootstraps,
2136         Follins_southBoots$bootstraps
2137     ),
2138     triangles = c(trisCenterBoot, trisSouthBoot),
2139     boundaryAlpha = 0.1,
2140     axesColors = c("black", "black", "black"),
2141     fogStyle = "none",
2142     background = "white",
2143     simplePoints = TRUE,
2144     colors = c(replicate(
2145         length(Follins_northBoots$bootstraps), "orange"
2146     ), replicate(
2147         length(Follins_southBoots$bootstraps), "blue"
2148     )),
2149     group = oriTrivialGroup
2150 )
2151
2152 # If you wish to save this figure, maximize the plot window on your
2153 screen before running this line. It will save to your working directory
2154 folder.
2155 afterMaximizingWindow("Boots_WestMt_CS_1.png", "Boots_WestMt_CS_2.png")
2156
2157 # Plot the bootstrap comparison in an Equal Area plot
2158 lineEqualAreaPlotTwo(c(
2159     lapply(Follins_centerBoots$bootstraps, function(s)
2160         s[1,]),
2161     lapply(Follins_centerBoots$bootstraps, function(s)
2162         s[2,])
2163 ),
2164 c(
2165     lapply(Follins_southBoots$bootstraps, function(s)
2166         s[1,]),
2167     lapply(Follins_southBoots$bootstraps, function(s)
2168         s[2,])
2169 ))
2170
2171
2172 APPENDIX 5: Ahsahka segment data
2173 Ahsahka segment foliation-lineation pairs

```

location	easting	northing	strike	dip	trend	plunge
13-TSL-32	553090	5153114	288	53	30	58
13-TSL-20	553117	5153753	288	62	55	58
12-TSL-15	553121	5152301	300	56	24	55
13-TSL-31	553359	5152857	308	56	355	52
13-TSL-7	553379	5154706	295	63	313	29
12-TSL-8	553387	5152784	297	48	16	47

13-TSL-39	553464	5154349	294	66	346	65
13-TSL-6	553548	5154710	315	60	324	27
12-TSL-24	554220	5159474	288	50	41	48
12-TSL-29	554942	5160955	284	52	62	41
12-TSL-19	554947	5155847	286	59	337	52
13-TSL-4	554979	5155062	292	56	13	63
12-TSL-31	555105	5160561	285	62	50	57
12-TSL-9	555453	5149141	295	54	53	50
13-TSL-37.1	555527	5150617	323	63	53	63
13-TSL-36	555543	5152055	309	75	350	53
12-TSL-33	555631	5160286	264	71	61	48
12-TSL-36	556012	5157321	303	52	9	49
13-TSL-10.1	556042	5157269	247	48	18	40
13-TSL-1	556201	5156192	315	51	24	48
13-TSL-38.1	556490	5154729	302	54	332	48
12-TSL-21	556647	5156041	303	52	355	45
13-TSL-14	556812	5156241	293	69	348	58
13-TSL-50.4	557395	5164863	52	67	71	39
13-TSL-27	557447	5155990	309	66	22	68
13-TSL-25	557532	5156097	306	51	19	52
13-TSL-41	557800	5163786	74	31	238	16
13-TSL-24	557836	5155990	327	61	5	42
13-TSL-23.1	557969	5155993	330	78	349	57
13-TSL-42	557971	5163945	39	38	60	9
13-TSL-43	558190	5164262	357	30	28	16
13-TSL-49.9	558304	5164892	206	42	214	5
13-TSL-46	558615	5169780	326	49	67	47
13-TSL-44.3	558729	5164569	21	41	41	15
13-TSL-44.32	558745	5164569	44	58	39	4
13-TSL-44.1	558786	5164589	15	33	53	12
13-TSL-49.4	558956	5168786	336	59	61	55
13-TSL-45.4	558972	5165164	290	61	70	49
13-TSL-49.5	558982	5168403	319	45	49	45
13-TSL-49	559063	5170431	317	38	33	35
13-TSL-49.1	559075	5170416	22	35	69	25
13-TSL-49.2	559077	5170414	335	52	65	52
13-TSL-49.3	559079	5170412	323	61	51	59
13-TSL-49.6	559117	5167347	311	55	70	51
13-TSL-46.1	559163	5170086	320	57	55	54
13-TSL-45.9	559193	5167930	327	50	46	45

13-TSL-45.8	559400	5167731	309	50	78	44
13-TSL-49.7	559479	5165587	325	34	37	33
13-TSL-48.2	559485	5170232	300	76	59	66
12-TSL-14	559665	5163392	279	30	48	23
13-TSL-44.6	559887	5165513	350	50	66	46
13-TSL-47i	559899	5170340	292	83	94	73
13-TSL-48	559987	5170419	283	86	56	68
13-TSL-45.7	560015	5167128	344	49	36	41
13-TSL-45.2	560039	5166529	11	30	48	19
13-TSL-52.2	561482	5161721	312	46	35	43
13-TSL-52.4	562182	5161385	329	40	30	36
13-TSL-52.3	562183	5161384	341	39	71	38
13-TSL-53.2	562445	5161179	335	35	69	34
13-TSL-53	562469	5161107	308	44	47	42
13-TSL-53.4	562495	5161175	318	49	59	47
13-TSL-54.1	563347	5160922	325	55	70	57
13-TSL-54.2	563387	5160922	305	58	94	39
13-TSL-55.1	563640	5161621	349	35	41	30
12-TSL-11	563916	5160475	300	60	79	48
13-TSL-58.1	564407	5161525	316	89	315	6
13-TSL-58	566295	5161832	326	61	74	57
13-TSL-57.1	566475	5161692	327	58	29	58
13-TSL-56.8	567079	5161942	347	58	7	33

2174

2175 **APPENDIX 6: Ahsahka segment statistical analysis**

2176 **### PRELIMINARY WORK ###**

2177 # Set the working directory.

2178 setwd("~/Desktop/20170620geologyGeometry")

2179 # Load the necessary R libraries.

2180 source("library/all.R")

2181 # NOTE: Run the following line only if you have compiled C. Markov chain
2182 Monte Carlo requires C. Load the necessary library

2183 source("libraryC/all.R")

2184 source("JSG_statsFunctions.r")

2185

2186 #=====LOAD

THE

2187 DATA=====

2188

2189 # Load the foliation-lineation data

2190 Follins <- geoDataFromFile("data/Follins_Ahs.csv")

2191

2192 # Check how many measurements there are

2193 nrow(Follins)

2194

```

2195 #=====PLOT
2196 DATA=====
2197
2198 # Plot foliation-lineation locations in map view
2199 plot(
2200     Follins$easting,
2201     Follins$northing,
2202     xlab = "Easting (meters)",
2203     ylab = "Northing (meters)"
2204 )
2205
2206 # Plot the data in Equal area plot. Pole to foliation (red), Lineation
2207 (cyan)
2208 lineEqualAreaPlotTwo(lapply(Follins$rotation, function(s)
2209     s[1, ]),
2210     lapply(Follins$rotation, function(s)
2211         s[2, ]))
2212
2213 # Plot Follins in Equal Area plots with Kamb contours (numbers are 2
2214 sigma values)
2215 # Poles to foliation (circles)
2216 lineKambPlot(lapply(Follins$rotation, function(s)
2217     s[1, ]), c(2, 6, 10, 14, 18))
2218 # Lineations (squares)
2219 lineKambPlot(lapply(Follins$rotation, function(s)
2220     s[2, ]),
2221     c(2, 6, 10, 14, 18),
2222     shapes = c("s"))
2223
2224 # Plot the Follins in an Equal Volume plot (after Davis and Titus, 2017)
2225 oriEqualVolumePlot(
2226     Follins$rotation,
2227     group = oriLineInPlaneGroup,
2228     backgroundColor = "white",
2229     curveColor = "black",
2230     boundaryAlpha = 0.1,
2231     colors = "black",
2232     axesColors = c("black", "black", "black"),
2233     fogStyle = "none"
2234 )
2235
2236 # If you wish to save this figure, maximize the plot window on your
2237 screen before running this line. It will save to your working directory
2238 folder.
2239 afterMaximizingWindow("Ahsahka_EqualVol1.png", "Ahsahka_EqualVol2.png")
2240
2241 # NOTE: Requires C. Skip if you have not compiled C. Plot 6-sigma Kamb
2242 contours for the data in an equal volume plot (after Davis and Titus,
2243 2017)
2244 oricKambPlot(
2245     Follins$rotation,
2246     group = oriLineInPlaneGroup,
2247     multiple = 6,

```

```

2248     backgroundColor = "white",
2249     curveColor = "black",
2250     boundaryAlpha = 0.1,
2251     colors = "black",
2252     axesColors = c("black", "black", "black"),
2253     fogStyle = "none"
2254 )
2255
2256 # If you wish to save this figure, maximize the plot window on your
2257 screen before running this line. It will save to your working directory
2258 folder.
2259 afterMaximizingWindow("Ahsahka_Kamb_equalVol1.png",
2260 "Ahsahka_Kamb_EqualVol2.png")
2261
2262
2263 #=====IDENTIFY GEOGRAPHIC
2264 TRENDS=====
2265
2266 # 1) Visually inspect the possibility of geographic trends in the data
2267
2268 # Plot foliation-lineation locations in map view, colored in gray-scale
2269 by northing. This shading will be used in the following plots.
2270 plot(
2271     Follins$easting,
2272     Follins$northing,
2273     col = shades(Follins$northing),
2274     pch = 19
2275 )
2276
2277 # Plot foliation-lineation orientations in an equal volume plot (after
2278 Davis and Titus, 2017)
2279 oriEqualVolumePlot(
2280     Follins$rotation,
2281     group = oriLineInPlaneGroup,
2282     simplePoints = FALSE,
2283     backgroundColor = "white",
2284     curveColor = "black",
2285     boundaryAlpha = 0.1,
2286     colors = shades(Follins$northing),
2287     axesColors = c("black", "black", "black"),
2288     fogStyle = "none"
2289 )
2290
2291 # If you wish to save this figure, maximize the plot window on your
2292 screen before running this line. It will save to your working directory
2293 folder.
2294 afterMaximizingWindow("Ahsahka_shadesNorthing_equalVol1.png",
2295 "Ahsahka_shadesNorthing_EqualVol2.png")
2296
2297 # Plot foliation-lineation orientations colored by northing. Poles to
2298 foliation (circles), Lineation (squares)
2299 lineEqualAreaPlot(
2300     c(

```



```

2301         lapply(Follins$rotation, function(s)
2302             s[1, ]),
2303         lapply(Follins$rotation, function(s)
2304             s[2, ]
2305         ),
2306         col = shades(Follins$northing),
2307         shapes = c(replicate(length(Follins$rotation), "c"),
2308 replicate(length(Follins$rotation), "s"))
2309     )
2310
2311     #-----
2312
2313     # 2) Run geodesic regressions to quantify any extant trends.
2314
2315     # Create a temporary version of the foliation lineation data and
2316     # establish a geographic criteria that encompasses all the data.
2317     FollinsAll <- Follins
2318     AhsAllCrit <-
2319         FollinsAll$easting > 550000 & FollinsAll$northing > 5140000
2320     FollinsAll$domain <- replicate(nrow(FollinsAll), 1)
2321     FollinsAll$domain[AhsAllCrit] <- 1
2322
2323     # 2A) Perform a geodesic regression with respect to azimuths every 10
2324     # degrees on all data, using the geographic criteria defined above. Each
2325     # regression asks the question: "does orientation change linearly with
2326     # respect to an azimuth towards x degrees"
2327     regressions <- regressionSweep(FollinsAll, 10, (AhsAllCrit), 0)
2328
2329     # Create a data frame that contains the summary information for the
2330     # regressions
2331     regressionsSum <- data.frame(regressions[[19]])
2332
2333     # Name the columns of the data frame
2334     names(regressionsSum) = c("Azimuth", "Error", "Min_Eigen", "R_squared",
2335     "Pvalue")
2336
2337     # View the summary table for the output.
2338     regressionsSum
2339
2340     # Plot the R^2 value as a function of Azimuth
2341     plot(
2342         as.vector(regressionsSum$Azimuth),
2343         as.vector(regressionsSum$R_squared),
2344         xlab = "Azimuth",
2345         ylab = "R-squared"
2346     )
2347
2348     #-----
2349
2350     # 3) Run kernal regression to quantify any extant trends.
2351
2352     # Recategorize the symmetric copies of the data.
2353     mu <- oriFrechetMean(Follins$rotation, group=oriLineInPlaneGroup)

```

```

2354 Follins$rotation <- oriNearestRepresentatives(Follins$rotation, mu,
2355 group=oriLineInPlaneGroup)
2356
2357 # Define the bandwidth for a kernel regression. If desired, uncomment the
2358 next line (it may take ~20 minutes to run). Otherwise, use the pre-
2359 calculated value, "bandwidth".
2360 bandwidth <-
2361 rotBandwidthForKernelRegression(Follins$northing, Follins$rotation)
2362 bandwidth <- 0.02314421
2363
2364 # Perform a kernel regression with respect to northing. The kernel
2365 function is analogous to finding a best fit curve in 2D.
2366 kernelReg <- lapply(
2367   seq(
2368     from = min(Follins$northing),
2369     to = max(Follins$northing),
2370     by = 100
2371   ),
2372   rotKernelRegression,
2373   Follins$northing,
2374   Follins$rotation,
2375   bandwidth,
2376   numSteps = 1000
2377 )
2378
2379 # Filter the regression results to only keep those that give an error
2380 of 0 and minimum eigen value > 0
2381 sapply(kernelReg, function(regr)
2382   regr$error)
2383 sapply(kernelReg, function(regr) {
2384   regr$minEigenvalue > 0
2385 })
2386
2387
2388 # Plot the regression curve in an equal volume plot.
2389 kernelRegCurve <- lapply(kernelReg, function(regr)
2390   regr$r)
2391 oriEqualVolumePlot(
2392   kernelRegCurve,
2393   group=oriLineInPlaneGroup,
2394   backgroundColor = "white",
2395   curveColor = "black",
2396   boundaryAlpha = 0.1,
2397   colors = "black",
2398   axesColors = c("black", "black", "black"),
2399   fogStyle = "none",
2400   simplePoints = TRUE
2401 )
2402
2403 # Compute the R^2 for the kernel regression.
2404 KernRsquared <-
2405   rotRsquaredForKernelRegression(Follins$northing, Follins$rotation,
2406   bandwidth, numSteps =

```

```

2407                                     1000)
2408
2409 # View the R^2 value
2410 KernRsquared
2411
2412 # Do a permutation test for significance. This takes a significant amount
2413 of time (>>1 hr)
2414 rSquareds <-
2415     rotKernelRegressionPermutations(Follins$northing, Follins$rotation,
2416     bandwidth, numPerms =
2417                                     1000)
2418     length(rSquareds)
2419 p <- sum(rSquareds > KernRsquared$rSquared)
2420 p
2421
2422
2423 #=====DEFINE GEOGRAPHIC
2424 DOMAINS=====
2425
2426 # Define geographic criteria, based on sampling area, with the proposed
2427 shear zone boundary into account.
2428 domain1Crit <- Follins$easting < 560000 & Follins$northing < 5158000
2429 domain2Crit <- (Follins$easting >= 560000 & Follins$northing < 5163000)
2430 | Follins$easting >= 565000
2431 domain3Crit <- !(domain1Crit | domain2Crit)
2432
2433 # # Create a new column in the dataframe in which to store the domain
2434 information
2435 Follins$domain <- replicate(nrow(Follins), 1)
2436
2437 # Classify the foliation-only dataset by domain
2438 Follins$domain[domain1Crit] <- 1
2439 Follins$domain[domain2Crit] <- 2
2440 Follins$domain[domain3Crit] <- 3
2441
2442 # Plot the locations of the foliation-lineation data in map view, each
2443 domain a different color. Domain 1 (Red), Domain 2 (Green), Domain 3
2444 (Blue)
2445 plot(
2446     x = Follins$easting,
2447     y = Follins$northing,
2448     xlab = "Easting (meters)",
2449     ylab = "Northing (meters)",
2450     col = hues(Follins$domain),
2451     pch = 19
2452 )
2453
2454 #=====IDENTIFY GEOGRAPHIC TRENDS WITHIN
2455 DOMAINS=====
2456
2457 # 1) Make sure each domain is roughly unimodal
2458 #DOMAIN 1, n = 23.
2459 oriEqualVolumePlot(

```

```

2460     Follins$rotation[domain1Crit],
2461     oriLineInPlaneGroup,
2462     backgroundColor = "white",
2463     curveColor = "black",
2464     boundaryAlpha = 0.1,
2465     colors = "black",
2466     axesColors = c("black", "black", "black"),
2467     fogStyle = "none"
2468 )
2469 oricKambPlot(
2470     Follins$rotation[domain1Crit],
2471     oriLineInPlaneGroup,
2472     backgroundColor = "white",
2473     curveColor = "black",
2474     boundaryAlpha = 0.1,
2475     colors = "black",
2476     axesColors = c("black", "black", "black"),
2477     fogStyle = "none"
2478 )
2479
2480 #DOMAIN 2, n = 14.
2481 oriEqualVolumePlot(
2482     Follins$rotation[domain2Crit],
2483     oriLineInPlaneGroup,
2484     backgroundColor = "white",
2485     curveColor = "black",
2486     boundaryAlpha = 0.1,
2487     colors = "black",
2488     axesColors = c("black", "black", "black"),
2489     fogStyle = "none"
2490 )
2491 oricKambPlot(
2492     Follins$rotation[domain2Crit],
2493     oriLineInPlaneGroup,
2494     backgroundColor = "white",
2495     curveColor = "black",
2496     boundaryAlpha = 0.1,
2497     colors = "black",
2498     axesColors = c("black", "black", "black"),
2499     fogStyle = "none"
2500 )
2501
2502 #DOMAIN 3, n = 32.
2503 oriEqualVolumePlot(
2504     Follins$rotation[domain2Crit],
2505     oriLineInPlaneGroup,
2506     backgroundColor = "white",
2507     curveColor = "black",
2508     boundaryAlpha = 0.1,
2509     colors = "black",
2510     axesColors = c("black", "black", "black"),
2511     fogStyle = "none"
2512 )

```

```

2513 oriEqualVolumePlot(Follins$rotation[domain2Crit],
2514                     oriLineInPlaneGroup,
2515                     col = hues(Follins$easting[domain2Crit]))
2516 oricKambPlot(
2517     Follins$rotation[domain2Crit],
2518     oriLineInPlaneGroup,
2519     backgroundColor = "white",
2520     curveColor = "black",
2521     boundaryAlpha = 0.1,
2522     colors = "black",
2523     axesColors = c("black", "black", "black"),
2524     fogStyle = "none"
2525 )
2526
2527
2528 # Plot all the foliation data in an equal volume plot (after Davis and
2529 Titus, 2017), colored by domain. Domain 1 (red), Domain 2 (green), Domain
2530 3 (blue)
2531 oriEqualVolumePlot(
2532     Follins$rotation,
2533     oriLineInPlaneGroup,
2534     col = hues(Follins$domain),
2535     backgroundColor = "white",
2536     curveColor = "black",
2537     boundaryAlpha = 0.2,
2538     axesColors = c("black", "black", "black"),
2539     fogStyle = "none"
2540 )
2541
2542 #-----
2543
2544 # 2) Look for geographic patterns in plots.
2545 #DOMAIN 1, n = 23. maybe a very faint rainbow--more like clumping of the
2546 dark blues together.
2547 oriEqualVolumePlot(Follins$rotation[domain1Crit],
2548                     oriLineInPlaneGroup,
2549                     col = shades(Follins$northing[domain1Crit]))
2550
2551 #DOMAIN 2, n = 14. Part of the problem with small sample sizes--you can't
2552 really know whether the datapoints are independent...
2553 oriEqualVolumePlot(Follins$rotation[domain2Crit],
2554                     oriLineInPlaneGroup,
2555                     col = shades(Follins$northing[domain2Crit]))
2556
2557 #DOMAIN 3, n = 32. No rainbow.
2558 oriEqualVolumePlot(Follins$rotation[domain2Crit],
2559                     oriLineInPlaneGroup,
2560                     col = shades(Follins$easting[domain2Crit]))
2561
2562 #-----
2563

```

```

2564 # 3) From visual inspection of the above plots, Domain 1 and Domain 3
2565 potentially have geographic dependency. Perform regressions on these two
2566 domains.
2567
2568 #DOMAIN 1, n = 23. Perform a series of Geodesic Regressions for every
2569 10° azimuth, with p-values not calculated.
2570 regressionsD1 <- regressionSweep(Follins, 10, domain1Crit, 0)
2571
2572 # Create a dataframe with the summary information for the regressions
2573 regressionsD1Sum <- data.frame(regressionsD1[[19]])
2574
2575 # Name the columns of the data frame
2576 names(regressionsD1Sum) = c("Azimuth", "Error", "Min_Eigen",
2577 "R_squared", "Pvalue")
2578
2579 # View the summary table for the output.
2580 regressionsD1Sum
2581
2582 # Plot the R^2 value as a function of Azimuth
2583 plot(
2584   as.vector(regressionsD1Sum$Azimuth),
2585   as.vector(regressionsD1Sum$R_squared),
2586   xlab = "Azimuth",
2587   ylab = "R-squared"
2588 )
2589
2590 #-----
2591
2592 # Define the Azimuth with the highest R-squared value. (In this case,
2593 130)
2594 azimuthD1 <-
2595   Follins$easting[domain1Crit] * sin(130 * degree) +
2596   Follins$northing[domain1Crit] *
2597   cos(130 * degree)
2598
2599 # DOMAIN 1, n = 23. Define the bandwidth for a kernel regression. If
2600 desired, uncomment the next line (it may take ~20 minutes to run).
2601 Otherwise, use the pre-calculated value, "bandwidthD1".
2602 #                               bandwidthD1 <-
2603 rotBandwidthForKernelRegression(azimuthD1, Follins$rotation[domain1Crit
2604 ])
2605 bandwidthD1 <- 0.02202349
2606
2607 #Run the kernel regression
2608 kernelRegD1 <- lapply(
2609   seq(
2610     from = min(azimuthD1),
2611     to = max(azimuthD1),
2612     by = 100
2613   ),
2614   rotKernelRegression,
2615   azimuthD1,
2616   Follins$rotation[domain1Crit],

```

```

2617     bandwidthD1,
2618     numSteps = 1000
2619 )
2620 sapply(kernelRegD1, function(regr)
2621     regr$error)
2622 sapply(kernelRegD1, function(regr) {
2623     regr$minEigenvalue > 0
2624 })
2625
2626 # Plot the regression curve
2627 kernelRegCurveD1 <- lapply(kernelRegD1, function(regr)
2628     regr$r)
2629 oriEqualVolumePlot(Follins$rotation[domain1Crit],
2630     curves = list(kernelRegCurveD1),
2631     simplePoints = TRUE, oriLineInPlaneGroup)
2632
2633 # Compute the R^2.
2634 KernRsquaredD1 <-
2635     rotRsquaredForKernelRegression(azimuthD1,
2636     Follins$rotation[domain1Crit], bandwidthD1, numSteps =
2637         1000)
2638 KernRsquaredD1
2639
2640 # Do a permutation test for significance.
2641 rSquaredsD1 <-
2642     rotKernelRegressionPermutations(azimuthD1,
2643     Follins$rotation[domain1Crit], bandwidthD1, numPerms =
2644         1000)
2645 length(rSquareds)
2646 pD1 <- sum(rSquareds > KernRsquared$rSquared)
2647
2648 #the p-value for this regression.
2649 pD1
2650
2651 #-----
2652
2653 #Now do the same thing for Domain 3
2654
2655 #DOMAIN 3, n = 32. A series of Geodesic Regressions for every 10° azimuth,
2656 with p-values not calculated.
2657 regressionsD3 <- regressionSweep(Follins, 10, domain3Crit, 0)
2658
2659 # Create a dataframe with the summary information for the regressions
2660 regressionsD3Sum <- data.frame(regressionsD3[[19]])
2661
2662 # Name the columns of the data frame
2663 names(regressionsD3Sum) = c("Azimuth", "Error", "Min_Eigen",
2664 "R_squared", "Pvalue")
2665
2666 # View the summary table for the output.
2667 regressionsD3Sum
2668
2669 # Plot the R^2 value as a function of Azimuth

```

```

2670 plot(
2671     as.vector(regressionsD3Sum$Azimuth),
2672     as.vector(regressionsD3Sum$R_squared),
2673     xlab = "Azimuth",
2674     ylab = "R-squared"
2675 )
2676
2677 #-----
2678
2679 # Define the Azimuth with the highest R-squared value. (In this case,
2680 60)
2681 azimuthD3 <-
2682     Follins$easting[domain3Crit] * sin(60 * degree) +
2683     Follins$northing[domain3Crit] *
2684     cos(60 * degree)
2685
2686 #DOMAIN 3, n = 32. Define the bandwidth for a kernel regression. If
2687 desired, uncomment the next line (it may take ~20 minutes to run).
2688 Otherwise, use the pre-calculated value, "bandwidthD3".
2689 bandwidthD3 <-
2690     rotBandwidthForKernelRegression(Follins$northing[domain3Crit],
2691     Follins$rotation[domain3Crit])
2692 bandwidthD3 <- 0.1331092
2693
2694 #Run the kernel regression
2695 kernelRegD3 <- lapply(
2696     seq(
2697         from = min(azimuthD3),
2698         to = max(azimuthD3),
2699         by = 100
2700     ),
2701     rotKernelRegression,
2702     azimuthD3,
2703     Follins$rotation[domain3Crit],
2704     bandwidthD3,
2705     numSteps = 1000
2706 )
2707 sapply(kernelRegD3, function(regr)
2708     regr$error)
2709 sapply(kernelRegD3, function(regr) {
2710     regr$minEigenvalue > 0
2711 })
2712
2713 # Plot the regression curve
2714 kernelRegCurveD3 <- lapply(kernelRegD3, function(regr)
2715     regr$r)
2716 oriEqualVolumePlot(Follins$rotation[domain3Crit],
2717     curves = list(kernelRegCurveD3),
2718     simplePoints = TRUE,
2719     oriLineInPlaneGroup)
2720
2721 # Compute the R^2.
2722 KernRsquaredD3 <-

```



```

2723     rotRsquaredForKernelRegression(azimuthD3,
2724 Follins$rotation[domain3Crit], bandwidthD3, numSteps = 1000)
2725 KernRsquaredD3
2726
2727 # Do a permutation test for significance.
2728 rSquaredsD3 <-
2729     rotKernelRegressionPermutations(Follins$northing[domain3Crit],
2730 Follins$rotation[domain3Crit], bandwidthD3, numPerms = 1000)
2731 length(rSquareds)
2732 pD3 <- sum(rSquareds > KernRsquaredD3$rSquared)
2733
2734 #the p-value for this regression.
2735 pD3
2736
2737 ##=====STATISTICAL
2738 DESCRIPTORS=====
2739
2740 # 1) Domain 1, n = 23
2741
2742 # Frechet mean (minimizes the Frechet variance).
2743 FrechetMeanDom1 <-
2744     oriFrechetMean(Follins$rotation[domain1Crit], oriLineInPlaneGroup)
2745 geoStrikeDipRakeDegFromRotation(FrechetMeanDom1)
2746 FrechetVarDom1 <-
2747     oriVariance(Follins$rotation[domain1Crit], FrechetMeanDom1,
2748 oriLineInPlaneGroup)
2749
2750 # Plot the FrechetMean in an Equal Angle plot
2751 FrechetCurvesDom1 <-
2752     lapply(Follins$rotation[domain1Crit], function(r)
2753         rotGeodesicPoints(FrechetMeanDom1, r, 10))
2754 oriEqualAnglePlot(points = Follins$rotation[domain1Crit], curves =
2755 FrechetCurvesDom1)
2756
2757 # Print the Strike, Dip, Rake of the Frechet mean.
2758 geoStrikeDipRakeDegFromRotation(FrechetMeanDom1)
2759
2760 # Print the Frechet variance.
2761 FrechetVarDom1
2762
2763 #-----
2764
2765 # 2) Domain 2, n = 14
2766 #Frechet mean minimizes the Frechet variance.
2767 FrechetMeanDom2 <-
2768     oriFrechetMean(Follins$rotation[domain2Crit], oriLineInPlaneGroup)
2769 geoStrikeDipRakeDegFromRotation(FrechetMeanDom2)
2770 FrechetVarDom2 <-
2771     oriVariance(Follins$rotation[domain2Crit], FrechetMeanDom2,
2772 oriLineInPlaneGroup)
2773
2774 # Plot the FrechetMean in an Equal Angle plot
2775 FrechetCurvesDom2 <-

```

```

2776     lapply(Follins$rotation[domain2Crit], function(r)
2777         rotGeodesicPoints(FrechetMeanDom2, r, 10))
2778 oriEqualAnglePlot(points = Follins$rotation[domain2Crit], curves =
2779 FrechetCurvesDom2)
2780
2781 # Print the Strike, Dip, Rake of the Frechet mean.
2782 geoStrikeDipRakeDegFromRotation(FrechetMeanDom2)
2783
2784 # Print the Frechet variance
2785 FrechetVarDom2
2786
2787 #-----
2788
2789 # 3) Domain 3, n = 32
2790 #Frechet mean minimizes the Frechet variance.
2791 FrechetMeanDom3 <-
2792     oriFrechetMean(Follins$rotation[domain3Crit], oriLineInPlaneGroup)
2793 geoStrikeDipRakeDegFromRotation(FrechetMeanDom3)
2794 FrechetVarDom3 <-
2795     oriVariance(Follins$rotation[domain3Crit], FrechetMeanDom3,
2796 oriLineInPlaneGroup)
2797
2798 # Plot the FrechetMean in an Equal Angle plot
2799 FrechetCurvesDom3 <-
2800     lapply(Follins$rotation[domain3Crit], function(r)
2801         rotGeodesicPoints(FrechetMeanDom3, r, 10))
2802 oriEqualAnglePlot(points = Follins$rotation[domain3Crit], curves =
2803 FrechetCurvesDom3)
2804
2805 # Print the Strike, Dip, Rake of the Frechet mean.
2806 geoStrikeDipRakeDegFromRotation(FrechetMeanDom3)
2807
2808 # Print the Frechet variance.
2809 FrechetVarDom3
2810
2811 #-----
2812
2813 # 4) Dispersion of the data
2814
2815 #The standard way to get at dispersion is to compute the maximum matrix
2816 fisher likelihood. The matrix fisher distribution comprises a kind of
2817 "mean" and a positive definite symmetric matrix, which characterises the
2818 anisotropic dispersion.
2819
2820 # Redefine the rotations to be within one of the symmetric copies
2821 mu <- oriFrechetMean(Follins$rotation, group = oriLineInPlaneGroup)
2822 Follins$rotation <-
2823     oriNearestRepresentatives(Follins$rotation, mu, group =
2824 oriLineInPlaneGroup)
2825
2826 # Northern Domain, n = 16. Fisher maximum likelihood
2827 mleDom1 <- rotFisherMLE(Follins$rotation[domain1Crit])
2828 mleDom1

```

```

2829 eigen(mleDom1$kHat, symmetric = TRUE, only.value = TRUE)$values
2830
2831 #Central Domain, n = 34. Fisher maximum likelihood
2832 mleDom2 <- rotFisherMLE(Follins$rotation[domain2Crit])
2833 mleDom2
2834 eigen(mleDom2$kHat, symmetric = TRUE, only.value = TRUE)$values
2835
2836 #Southern Domain, n = 79. Fisher maximum likelihood
2837 mleDom3 <- rotFisherMLE(Follins$rotation[domain3Crit])
2838 mleDom3
2839 eigen(mleDom3$kHat, symmetric = TRUE, only.value = TRUE)$values
2840
2841 #=====INFERENCE=====
2842 =====
2843
2844 # Here, we can perform some statistical techniques to use information
2845 about the samples (each domain) to compute a probability cloud of the
2846 mean of the population(s) from which those samples were taken.
2847 # From numerical work in Davis and Titus (2017), MCMC will work the best
2848 for small sample sizes that have the matrix fisher anisotropy
2849 (eigenvalues) of (large, large, small). All four domains have this
2850 anisotropy, and have too few data points to use the (computationally
2851 quicker) bootstrapping method.
2852
2853 #We'll do both and compare.
2854
2855 # Compute the bootstrap cloud of means
2856 Follins_domain1Boots <-
2857     oriBootstrapInference(Follins$rotation[domain1Crit],      10000,
2858     oriLineInPlaneGroup)
2859 Follins_domain2Boots <-
2860     oriBootstrapInference(Follins$rotation[domain2Crit],      10000,
2861     oriLineInPlaneGroup)
2862 Follins_domain3Boots <-
2863     oriBootstrapInference(Follins$rotation[domain3Crit],      10000,
2864     oriLineInPlaneGroup)
2865
2866 # Compute the Markov chain Monte Carlo cloud of means
2867 Follins_domain1MCMC <-
2868
2869     oricWrappedTrivariateNormalMCMCInference(Follins$rotation[domain1Crit]
2870     ,
2871     group =
2872     oriLineInPlaneGroup,
2873     numCollection = 100)
2874 Follins_domain2MCMC <-
2875
2876     oricWrappedTrivariateNormalMCMCInference(Follins$rotation[domain2Crit]
2877     ,
2878     group =
2879     oriLineInPlaneGroup,
2880     numCollection = 100)
2881 Follins_domain3MCMC <-

```

```

2882
2883 oricWrappedTrivariateNormalMCMCInference(Follins$rotation[domain3Crit]
2884 ,
2885                                     group                                =
2886 oriLineInPlaneGroup,
2887                                     numCollection = 100)
2888
2889 #-----
2890
2891 # 1) Northern v. Central domains
2892
2893 # A) Using MCMC
2894
2895 # Construct the 95% confidence ellipsoids from small triangles
2896 tris1_MCMC <-
2897     rotEllipsoidTriangles(
2898         Follins_domain1MCMC$mBar,
2899         Follins_domain1MCMC$leftCovarInv,
2900         Follins_domain1MCMC$q095,
2901         numNonAdapt = 4
2902     )
2903 tris2_MCMC <-
2904     rotEllipsoidTriangles(
2905         Follins_domain2MCMC$mBar,
2906         Follins_domain2MCMC$leftCovarInv,
2907         Follins_domain2MCMC$q095,
2908         numNonAdapt = 4
2909     )
2910
2911 # Plot the MCMC comparison in an equal Volume plot, with 95% confidence
2912 ellipsoids
2913 oriEqualAnglePlot(
2914     points = c(Follins_domain1MCMC$ms, Follins_domain2MCMC$ms),
2915     boundaryAlpha = .1,
2916     axesColors = c("black", "black", "black"),
2917     fogStyle = "none",
2918     background = "white",
2919     triangles = c(tris1_MCMC, tris2_MCMC),
2920     simplePoints = TRUE,
2921     colors = c(replicate(length(
2922         Follins_domain1MCMC$ms
2923     ), "black"), replicate(
2924         length(Follins_domain2MCMC$ms), "orange"
2925     )),
2926     group = oriTrivialGroup
2927 )
2928
2929 # If you wish to save this figure, maximize the plot window on your
2930 screen before running this line. It will save to your working directory
2931 folder.
2932 afterMaximizingWindow("MCMC_Ahs_dld2_1.png", "MCMC_WestMt_dld2_2.png")
2933
2934 # Plot the MCMC comparison in an Equal Area plot

```

```

2935 lineEqualAreaPlotTwo(c(
2936     lapply(Follins_domain1MCMC$ms, function(s)
2937         s[1, ]),
2938     lapply(Follins_domain1MCMC$ms, function(s)
2939         s[2, ]))
2940 ),
2941 c(
2942     lapply(Follins_domain2MCMC$ms, function(s)
2943         s[1, ]),
2944     lapply(Follins_domain2MCMC$ms, function(s)
2945         s[2, ]))
2946 ))
2947
2948 #.....
2949
2950 # B) Using bootstrapping
2951
2952 # Construct the 95% confidence ellipsoids from small triangles
2953 tris1_Boot <-
2954     rotEllipsoidTriangles(
2955         Follins_domain1Boots$center,
2956         Follins_domain1Boots$leftCovarInv,
2957         Follins_domain1Boots$q095,
2958         numNonAdapt = 4
2959     )
2960 tris2_Boot <-
2961     rotEllipsoidTriangles(
2962         Follins_domain2Boots$center,
2963         Follins_domain2Boots$leftCovarInv,
2964         Follins_domain2Boots$q095,
2965         numNonAdapt = 4
2966     )
2967
2968 # Plot the bootstrap comparison in an equal Volume plot, with 95%
2969 confidence ellipsoids
2970 rotEqualAnglePlot(
2971     points = c(
2972         Follins_domain1Boots$bootstraps,
2973         Follins_domain2Boots$bootstraps
2974     ),
2975     triangles = c(tris1_Boot, tris2_Boot),
2976     boundaryAlpha = .1,
2977     axesColors = c("black", "black", "black"),
2978     fogStyle = "none",
2979     background = "white",
2980     simplePoints = TRUE,
2981     colors = c(replicate(
2982         length(Follins_domain1Boots$bootstraps), "black"
2983     ), replicate(
2984         length(Follins_domain2Boots$bootstraps), "orange"
2985     ))
2986 )
2987

```

```

2988 # If you wish to save this figure, maximize the plot window on your
2989 screen before running this line. It will save to your working directory
2990 folder.
2991 afterMaximizingWindow("Boots_Ahs_dld2_1.png", "Boots_Ahs_dld2_2.png")
2992
2993 # Plot the bootstrap comparison in an Equal Area plot
2994 lineEqualAreaPlotTwo(c(
2995     lapply(Follins_domain1Boots$bootstraps, function(s)
2996         s[1, ]),
2997     lapply(Follins_domain1Boots$bootstraps, function(s)
2998         s[2, ]),
2999 ),
3000 c(
3001     lapply(Follins_domain2Boots$bootstraps, function(s)
3002         s[1, ]),
3003     lapply(Follins_domain2Boots$bootstraps, function(s)
3004         s[2, ]),
3005 ))
3006
3007 #-----
3008
3009 # 2) Northern vs. Southern domains
3010
3011 # A) Using MCMC
3012
3013 # Construct the 95% confidence ellipsoids from small triangles
3014 tris1_MCMC <-
3015     rotEllipsoidTriangles(
3016         Follins_domain1MCMC$mBar,
3017         Follins_domain1MCMC$leftCovarInv,
3018         Follins_domain1MCMC$q095,
3019         numNonAdapt = 5
3020     )
3021 tris3_MCMC <-
3022     rotEllipsoidTriangles(
3023         Follins_domain3MCMC$mBar,
3024         Follins_domain3MCMC$leftCovarInv,
3025         Follins_domain3MCMC$q095,
3026         numNonAdapt = 5
3027     )
3028
3029 # Plot the bootstrap comparison in an equal Volume plot, with 95%
3030 confidence ellipsoids
3031 oriEqualAnglePlot(
3032     points = c(Follins_domain1MCMC$ms, Follins_domain3MCMC$ms),
3033     triangles = c(tris1_MCMC, tris3_MCMC),
3034     boundaryAlpha = 0.1,
3035     axesColors = c("black", "black", "black"),
3036     fogStyle = "none",
3037     background = "white",
3038     simplePoints = TRUE,
3039     colors = c(replicate(length(
3040         Follins_domain1MCMC$ms

```

```

3041     ), "black"), replicate(length(
3042         Follins_domain3MCMC$ms
3043     ), "blue")),
3044     group = oriTrivialGroup
3045 )
3046
3047 # If you wish to save this figure, maximize the plot window on your
3048 screen before running this line. It will save to your working directory
3049 folder.
3050 afterMaximizingWindow("MCMC_Ahs_dld3_1.png", "MCMC_Ahsdld3_2.png")
3051
3052 # Plot the MCMC comparison in an Equal Area plot
3053 lineEqualAreaPlotTwo(c(
3054     lapply(Follins_domain1MCMC$ms, function(s)
3055         s[1, ]),
3056     lapply(Follins_domain1MCMC$ms, function(s)
3057         s[2, ]
3058 ),
3059 c(
3060     lapply(Follins_domain3MCMC$ms, function(s)
3061         s[1, ]),
3062     lapply(Follins_domain3MCMC$ms, function(s)
3063         s[2, ]
3064 ))
3065
3066 #.....
3067
3068 # B) Using bootstrapping
3069 # Construct the 95% confidence ellipsoids from small triangles
3070 tris1_Boot <-
3071     rotEllipsoidTriangles(
3072         Follins_domain1Boots$center,
3073         Follins_domain1Boots$leftCovarInv,
3074         Follins_domain1Boots$q095,
3075         numNonAdapt = 4
3076     )
3077 tris3_Boot <-
3078     rotEllipsoidTriangles(
3079         Follins_domain3Boots$center,
3080         Follins_domain3Boots$leftCovarInv,
3081         Follins_domain3Boots$q095,
3082         numNonAdapt = 4
3083     )
3084
3085 # Plot the bootstrap comparison in an equal Volume plot, with 95%
3086 confidence ellipsoids.
3087 rotEqualAnglePlot(
3088     points = c(
3089         Follins_domain1Boots$bootstraps,
3090         Follins_domain3Boots$bootstraps
3091     ),
3092     triangles = c(tris1_Boot, tris3_Boot),
3093     boundaryAlpha = .1,

```

```

3094     axesColors = c("black", "black", "black"),
3095     fogStyle = "none",
3096     background = "white",
3097     simplePoints = TRUE,
3098     colors = c(replicate(
3099         length(Follins_domain1Boots$bootstraps), "black"
3100     ),
3101     replicate(
3102         length(Follins_domain3Boots$bootstraps), "blue"
3103     ))
3104 )
3105
3106 # If you wish to save this figure, maximize the plot window on your
3107 # screen before running this line. It will save to your working directory
3108 # folder.
3109 afterMaximizingWindow("Boots_Ahs_dld3_1.png", "Boots_Ahs_dld3_2.png")
3110
3111 # Plot the bootstrap comparison in an Equal Area plot
3112 lineEqualAreaPlotTwo(c(
3113     lapply(Follins_domain1Boots$bootstraps, function(s)
3114         s[1, ]),
3115     lapply(Follins_domain1Boots$bootstraps, function(s)
3116         s[2, ])
3117 ),
3118 c(
3119     lapply(Follins_domain3Boots$bootstraps, function(s)
3120         s[1, ]),
3121     lapply(Follins_domain3Boots$bootstraps, function(s)
3122         s[2, ])
3123 ))
3124
3125
3126
3127 #-----
3128
3129 # 3) Central vs. Southern domains
3130
3131 # A) Using MCMC
3132
3133 # Construct the 95% confidence ellipsoids from small triangles
3134 tris2_MCMC <-
3135     rotEllipsoidTriangles(
3136         Follins_domain2MCMC$mBar,
3137         Follins_domain2MCMC$leftCovarInv,
3138         Follins_domain2MCMC$q095,
3139         numNonAdapt = 4
3140     )
3141 tris3_MCMC <-
3142     rotEllipsoidTriangles(
3143         Follins_domain3MCMC$mBar,
3144         Follins_domain3MCMC$leftCovarInv,
3145         Follins_domain3MCMC$q095,
3146         numNonAdapt = 4

```



```

3147     )
3148
3149 # Plot the bootstrap comparison in an equal Volume plot, with 95%
3150 confidence ellipsoids
3151 oriEqualAnglePlot(
3152     points = c(Follins_domain2MCMC$ms, Follins_domain3MCMC$ms),
3153     triangles = c(tris2_MCMC, tris3_MCMC),
3154     boundaryAlpha = 0.1,
3155     axesColors = c("black", "black", "black"),
3156     fogStyle = "none",
3157     background = "white",
3158     simplePoints = TRUE,
3159     colors = c(replicate(
3160         length(Follins_domain2MCMC$ms), "orange"
3161     ), replicate(length(
3162         Follins_domain3MCMC$ms
3163     ), "blue")),
3164     group = oriTrivialGroup
3165 )
3166
3167 # If you wish to save this figure, maximize the plot window on your
3168 screen before running this line. It will save to your working directory
3169 folder.
3170 afterMaximizingWindow("MCMC_Ahs_d2d3_1.png", "MCMC_Ahs_d2d3_2.png")
3171
3172 # Plot the MCMC comparison in an Equal Area plot
3173 lineEqualAreaPlotTwo(c(
3174     lapply(Follins_domain2MCMC$ms, function(s)
3175         s[1, ]),
3176     lapply(Follins_domain2MCMC$ms, function(s)
3177         s[2, ]),
3178 ),
3179 c(
3180     lapply(Follins_domain3MCMC$ms, function(s)
3181         s[1, ]),
3182     lapply(Follins_domain3MCMC$ms, function(s)
3183         s[2, ]),
3184 ))
3185
3186 #.....
3187
3188
3189 # B) Using bootstrapping
3190
3191 # Construct the 95% confidence ellipsoids from small triangles
3192 tris2_Boot <-
3193     rotEllipsoidTriangles(
3194         Follins_domain2Boots$center,
3195         Follins_domain2Boots$leftCovarInv,
3196         Follins_domain2Boots$q095,
3197         numNonAdapt = 4
3198     )
3199 tris3_Boot <-

```

```

3200     rotEllipsoidTriangles(
3201         Follins_domain3Boots$center,
3202         Follins_domain3Boots$leftCovarInv,
3203         Follins_domain3Boots$q095,
3204         numNonAdapt = 4
3205     )
3206
3207 # Plot the bootstrap comparison in an equal Volume plot, with 95%
3208 confidence ellipsoids
3209 oriEqualAnglePlot(
3210     points = c(
3211         Follins_domain2Boots$bootstraps,
3212         Follins_domain3Boots$bootstraps
3213     ),
3214     triangles = c(tris2_Boot, tris3_Boot),
3215     boundaryAlpha = 0.1,
3216     axesColors = c("black", "black", "black"),
3217     fogStyle = "none",
3218     background = "white",
3219     simplePoints = TRUE,
3220     colors = c(replicate(
3221         length(Follins_domain1Boots$bootstraps), "orange"
3222     ), replicate(
3223         length(Follins_domain3Boots$bootstraps), "blue"
3224     )),
3225     group = oriTrivialGroup
3226 )
3227
3228 # If you wish to save this figure, maximize the plot window on your
3229 screen before running this line. It will save to your working directory
3230 folder.
3231 afterMaximizingWindow("Boots_Ahs_d2d3_1.png", "Boots_Ahs_d2d3_2.png")
3232
3233 # Plot the bootstrap comparison in an Equal Area plot
3234 lineEqualAreaPlotTwo(c(
3235     lapply(Follins_domain2Boots$bootstraps, function(s)
3236         s[1, ]),
3237     lapply(Follins_domain2Boots$bootstraps, function(s)
3238         s[2, ]),
3239 ),
3240 c(
3241     lapply(Follins_domain3Boots$bootstraps, function(s)
3242         s[1, ]),
3243     lapply(Follins_domain3Boots$bootstraps, function(s)
3244         s[2, ]),
3245 ))
3246
3247
3248
3249

```