

**Explorations of the lineup protocol for visual inference: application to high
dimension, low sample size problems and metrics to assess the quality.**

by

Niladri Roy Chowdhury

A thesis submitted to the graduate faculty
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Statistics

Program of Study Committee:

Dianne Cook, Major Professor

Heike Hofmann

Arka Ghosh

Peng Liu

Eric Cooper

Iowa State University

Ames, Iowa

2014

Copyright © Niladri Roy Chowdhury, 2014. All rights reserved.

DEDICATION

I would like to dedicate this dissertation to my parents, who always inspired me to pursue higher studies.

TABLE OF CONTENTS

LIST OF TABLES	vi
LIST OF FIGURES	viii
ACKNOWLEDGEMENTS	xv
ABSTRACT	xvi
CHAPTER 1. INTRODUCTION	1
1.1 Background	1
1.2 Review of Hypothesis Testing	2
1.3 Introduction to Visual Inference	5
1.3.1 Protocols of Visual Inference	5
1.4 Overview	10
1.5 Scope of my Research	11
CHAPTER 2. USING VISUAL STATISTICAL INFERENCE TO BETTER UNDERSTAND RANDOM CLASS SEPARATIONS IN HIGH DIMENSION, LOW SAMPLE SIZE DATA	13
2.1 Introduction	14
2.2 Visual inference methods	16
2.3 Dimension reduction	19
2.4 Amazon Turk experiments	22
2.5 Wasps application	22
2.6 Follow-up simulation experiment	25

2.6.1	Experimental design	25
2.6.2	Simulation process	26
2.6.3	Producing lineups	27
2.6.4	Data collection	28
2.7	Results	31
2.7.1	Effect of experimental factors on detection rate	31
2.7.2	Time taken to respond under different treatments	33
2.7.3	What affects decisions?	34
2.7.4	How do the null plots affect choices?	35
2.8	Conclusions	36

CHAPTER 3. UTILIZING DISTANCE METRICS ON LINEUPS TO EXAM-

	INE WHAT PEOPLE READ FROM DATA PLOTS	39
3.1	Introduction	40
3.2	Experimental Data	44
3.3	Null Generating Mechanism	44
3.4	Distance Measures	46
3.5	Distance Metric Distribution	51
3.6	Effect of Plot Type and Question of Interest	55
3.7	Metric Evaluation	56
3.8	Results	57
3.8.1	Turk Experiment 1 – Side by Side Boxplots	57
3.8.2	Turk Experiment 2 – Scatterplots with an Overlaid Regression Line	63
3.8.3	Turk Experiment 7 – Large p , Small n Data	67
3.9	Conclusion	73

CHAPTER 4. NULLABOR: AN R PACKAGE FOR VISUAL STATISTICAL

	INFERENCE	75
4.1	Introduction	76

4.2	Null generating mechanisms	77
4.2.1	Generate null data with a specific distribution	77
4.2.2	Generate null data by permuting a variable	78
4.2.3	Generate null data with null residuals from a model	78
4.3	Protocols	79
4.3.1	The lineup protocol	79
4.3.2	The Rorschach protocol	81
4.4	Distance metrics	83
4.4.1	Distance for univariate data	85
4.4.2	Distance based on regression parameters	85
4.4.3	Distance based on boxplots	85
4.4.4	Distance based on separation	86
4.4.5	Binned Distance	87
4.5	Calculation of mean distances and difference	87
4.5.1	Calculating the mean distances for the plots in the lineup	87
4.5.2	Calculating difference measure for lineups	89
4.6	Selection of bins	90
4.7	Distribution of distance metrics	91
4.7.1	Empirical distribution of distance metrics	92
4.7.2	Plotting the empirical distribution of the distance metric	93
CHAPTER 5. GENERAL CONCLUSIONS		97
5.1	General discussion	97
5.2	Possibilities for future research	98
APPENDIX A. SUPPLEMENTARY MATERIALS OF CHAPTER 2		99
A.1	Solutions to the Lineups	99
A.2	Choice of dimensions	99
BIBLIOGRAPHY		104

LIST OF TABLES

1.1	Comparison of visual inference with traditional hypothesis testing.	9
2.1	Comparison of visual inference with traditional hypothesis testing. Starting with the same hypothesis, the test statistic in a conventional setting is a real number while in visual inference it is a plot of the observed data. In conventional testing the value of the test statistic is compared with all possible values of the sampling distribution. H_o is rejected if it is extreme. In visual inference, the plot of the data is compared with a finite number of samples drawn from the null distribution. If the actual data plot is identifiable, then the null hypothesis is rejected.	18
2.2	Results of the Turk study on the wasps data. Detection rate for each lineup is shown, with the number of subjects, and p -value associated. The detection rate is highest for one of the purely noise lineups, which occurred because the plot with the most difference between groups happened to be the one that is randomly generated as the “real” data. Averaging the p -values for each set of lineups, for the wasps is 1.0, and for the pure noise, is 0.67 suggesting that the apparent separation in the wasp data (Toth et al. (2010)) is consistent with pure noise induced by the high dimensions.	24
2.3	Levels of the factors used for the simulation experiment.	26

2.4	Table summarizing results of experiment. Columns correspond to the estimate, the standard error and the p -value of the parameters used in logistic regression model. As dimension (p) increases, detection of separation decreases. Subjects can detect the separation if it exists even when $p = 100$. Subjects were equally good in 1D or 2D projections.	33
3.1	Overview of the different Turk experiments, from where data was taken to study distance metrics and how subjects read the plots.	45
A.1	Numerical summaries of dimension p for each value of δ . As the common region δ increases, the median dimension required to obtain the region increases.	103

LIST OF FIGURES

1.1	Decision regions for classical inference for $H_0 : \mu = \mu_0$ vs $H_a : \mu > \mu_0$	4
1.2	A typical lineup plot ($m = 20$) for testing $H_0 : \mu_1 = \mu_2$. When the alternative hypothesis is true the observed plot should have the largest vertical difference between the centers. Can you identify the observed plot?	7
1.3	Sampling distribution of the test statistic with the observed value and the values for the null plots corresponding to the lineup in Figure 2.2.	8
2.1	LD1 versus LD2 from an LDA on a randomly selected subset of 40 significantly different oligos : F, Foundress; G, gyne; Q, queen and W, worker. It can be noticed that the groups F and G are separated. This plot is generated to match Figure 2 in Toth et al. (2010).	15
2.2	A typical lineup ($m = 20$) for testing $H_o : \mu_1 = \mu_2$. When the alternative hypothesis is true, the observed data plot should have the largest vertical difference between the centers. Can you identify the observed data plot? The solution to the lineup is provided in the Appendix.	20
2.3	Lineup of the wasps data. One plot shows the observed data and the remaining 19 show null data where the wasp type labels were randomly assigned. Each plot was produced by conducting LDA on the 40D data to produce the 2D projection with best separation. Which plot shows the most separation between the 4 groups? The solution is provided in the Appendix.	23

2.4	Plots showing example 1D projections for $p = 2, 8, 15, 22, 28$ and $n = 30$ for purely noise data. The probability of obtaining a projection where groups are separated is calculated and displayed below the plots. Of course, once a projection is computed the groups are either separated or not - the event occurred or didn't - and we can see that the last two plots display separated groups. The difference between the groups increases as p increases, and the likelihood of obtaining a projection with separation increases.	25
2.5	The visual test statistics $V_1(\mathbf{Y})$ and $V_2(\mathbf{Y})$ used. $V_1(\mathbf{Y})$ is a horizontal jittered dot plot while $V_2(\mathbf{Y})$ is a scatterplot of the first and second dimensional projections, with color representing groups in both cases.	28
2.6	Lineup ($m = 20$) from treatment with $p = 20$, separation = Yes and $d = 1$. The subjects were asked to identify the plot with the most separated colors. Can you identify the observed data plot? The solution to the lineup is provided in the Appendix.	29
2.7	Lineup ($m = 20$) from treatment with $p = 100$, separation = No and $d = 2$. The subjects were asked to identify the plot with the most separation between the colored groups. Can you identify the observed data plot? The solution is provided in the Appendix.	30
2.8	Detection rate by dimension, faceted by projection and separation. The three points represents the three replicates for each treatment level. A fixed effects logistic regression model is overlaid on the points. It can be seen that the detection rate decreases as p increases for data with real separation. When the data is purely noise data, the detection rate is flat across dimensions. Detection rate does not change with projection. Even with $p = 100$ subjects more often detected separation than would be expected by chance.	32

2.9	Time taken in seconds to respond on log scale against dimension colored by separation and faceted by projection. A line shows the trend over dimension for each separation within projection. Bootstrap resampling bands are drawn for each colored lines. Time taken to respond is higher when the data has no separation. Also as dimension increases, the time to answer when there is separation is equal to the time taken when there is no separation.	34
2.10	Comparing the choices that subjects make for each lineup. Relative frequency of plots chosen against a measure of the average separation between groups, the larger the value the more separated are the groups. Each cell here shows the data for one of the lineups used in the experiment, 60 in total, and each “pin” represents a plot in the lineup, 20 for each lineup. Red indicates the observed data plot. Subjects are asked to pick the plot in the lineup where the groups are the most separated, so we would expect that more subjects would pick the plots with the largest average separation. In general, this happens, the tallest pins are in the right of each cell. The top three rows show the results for the data with separation, so the observed data plot (red) is typically the pin on the very left of the cell, less so for the higher dimensions which are the cells at right. Figure (a) shows 1D projections and Figure (b) shows for 2D projections. There is not much difference between the two figures.	35

2.11	Detection rate and mean time taken to respond in seconds are plotted against the difference for 1D and 2D projections separately. The difference is between maximum separation of all the null plots and separation of the observed data plot for each lineup for 1D projections but for 2D projections the difference is based on the average separation between the groups. The vertical line represents difference equal to 1 when the average separation of the observed data plot is equal to the maximum average separation of the null plots for 2D projection. The points left to the line indicates a difficult lineup in the sense that at least one of the null plots had a lower average separation value than the observed data plot. (a) and (b) As difference increases, detection rate increases. (c) and (d) As difference increases, mean time taken decreases indicating that the subjects have an easier time in identifying the observed data plot.	37
3.1	Lineup plot of size $m = 20$ using scatterplots with regression line overlaid. This tests $H_o : \beta_k = 0$, where covariate X_k is continuous. One of the plots in the lineup is the plot of the true data. The other plots are null plots generated by simulating data from a null model that assumes that the null hypothesis is true. Can you identify the plot with the steepest slope? . . .	42
3.2	If the lineup protocol was to be used instead of classical inference this is what it would look like. (a) Decision region (shaded in red) for classical inference for $H_0 : \mu = \mu_0$ vs $H_a : \mu > \mu_0$ and (b) values corresponding to the true value (red) and the null plots (blue) in a single lineup of size $m = 20$ that would be used to test the same null hypothesis. The actual data plot is extreme relative to the null plots, and observers would likely be able to pick it out, resulting in a decision to reject the null hypothesis. In practice, the lineup protocol would not be used if a classical test can be used.	43

3.3	Illustration of binned distance, for data with strong association (a), and the same data where one variable has been permuted (b). The scatterplot of the data is shown (left) along with the binned view of the data (center) and the number of points in each cell (right). Binned distance is the euclidean distance of these counts. The binned distance between these plots is 6.4807.	48
3.4	Illustration of three different distance metrics based on separation. Two dimensional projections are plotted with 3 groups. Minimum Separation (in (a)) calculates the minimum distance between points of each cluster from the other clusters. Average separation (in (b)) calculates the average distance of each point in a cluster to the other clusters. In (c), the cluster mean distance calculates the distance between the means of each cluster.	51
3.5	Optional caption for list of figures	53
3.6	Optional caption for list of figures	54
3.7	Optional caption for list of figures	56
3.8	An example lineup from Turk Experiment 1. The lineup has $m = 20$ plots of which one is the observed data plot and the remaining $m - 1$ are the null plots generated assuming that the null hypothesis is true. Subjects were asked to identify the plot which has the largest vertical difference between the two groups. Can you identify the observed data plot?	59
3.9	Optional caption for list of figures	60
3.10	Optional caption for list of figures	61
3.11	Optional caption for list of figures	62
3.12	An example lineup from Turk Experiment 2. In this lineup, one of the plots is the observed plot and the other 19 plots are the null plots generated assuming that the null hypothesis $H_o : \beta = 0$ is true. Subjects were asked to identify the plot with the steepest slope. Can you identify the observed plot ?	64
3.13	Optional caption for list of figures	65
3.14	Optional caption for list of figures	66

3.15	Optional caption for list of figures	68
3.16	An example lineup from Turk Experiment 7. Here two dimensional projections of the PDA index are plotted for a data with separation having $p = 20$ dimensions and $n = 30$ observations. The subjects were asked to identify the plot with the most separated colors. Can you identify the observed data plot?	69
3.17	Optional caption for list of figures	70
3.18	Optional caption for list of figures	71
3.19	Optional caption for list of figures	72
4.1	A typical lineup. The plot of the actual data is embedded in a set of null plots which are generated assuming that the null hypothesis is true. The variable “mpg” is permuted 19 times to obtain the 19 null plots. Can you identify the plot which has the steepest slope? The true position of the plot can be obtain by copying and pasting “decrypt(…)” from the output into R console.	82
4.2	Illustration of the Rorschach protocol. These nine scatterplots are obtained by permuting the variable “mpg” while keeping the variable “wt” fixed. Does it look like the strength of the relationship is same in all the plots? Is there a plot which has more interesting pattern than others?	84
4.3	Lineup plot of size $m = 20$ of scatterplots between “mpg” and “wt” with a regression line overlaid. The plot of the actual data is randomly placed among the set of null plots. The position of the plot of the actual data is 10.	88
4.4	Illustration of the optimum bin selection procedure. The obtained lineup data from subsection 4.5.1 is used to generate difference between the true plot and the maximum of the null plots using a binned distance for all combinations of number of bins on both x and y axes. The number of bins on x and y axes are plotted on x and y respectively with the colors showing the differences. The darker color represents larger difference.	91

4.5	Distribution of the regression based distance with the distance for the null plots of the lineup represented in black and the distance for the plot of the actual data represented by orange. The distance for the actual plot is much higher than the null plots and the actual plot can be easily identified as seen in Figure 4.3.	94
4.6	Distribution of the binned distance with the distance for the null plots of the lineup represented in black and the distance for the plot of the actual data represented by orange. The number of bins used is 6 on both axes. The distance for the actual plot falls within the distances for the null plots. This shows that the lineup is difficult in the sense that the actual cannot be easily identified. Hence the selection of the number of bins is important for binned distance.	95
4.7	Distribution of the binned distance with the distance for the null plots of the lineup represented in black and the distance for the plot of the actual data represented by orange. The number of bins used is 3 on x-axis and 2 on y-axis. The number of bins are selected by the optimal bin selection method using Figure 4.4. The distance for the actual plot is larger than the distances for the null plots which matches the lineup.	96
A.1	Plot showing the distribution of the sum of absolute difference of means for data with and without separation for different dimensions. The distributions of data with real separation (V) and purely noise data (U) are shown in brown and green respectively with the dark purple line showing the 5th percentile of V. The dark purple area shows the area of U which is greater than the 5th percentile of V. The dark purple region (δ) increases as dimension (p) increases.	102

ACKNOWLEDGEMENTS

First, I would like to thank my wife, Rianka, for her constant support, patience, understanding and encouragement throughout my graduate education. I would also like to thank my younger brother, Upamanyu, who have always supported and stood by me in my low moments.

I would like to thank my major professor, Dr. Dianne Cook for her valuable guidance and support. The various statistical methods learnt from her, specially on data visualizations and academic writing, will help me excel in my future endeavors. But her sense of humor, calmness and balance in life have had the biggest effects on me and it would help me to improve as a human being.

The suggestions provided by Dr. Heike Hofmann have helped improving the quality and scope of my research work. I am very thankful to her.

I really appreciate the valuable comments and suggestions made by my committee members. I would also like to thank them for their time and patience.

Graphics Working Group has provided me various suggestions and participated in a number of pilot studies. I would like to thank everyone in the group. I would specially thank Mahbubul Majumder for helping me whenever I needed.

I would also like to thank the Department of Statistics in Iowa State University for providing me this opportunity to pursue my doctorate degree. The inspiring teaching styles of many professors have helped me understand and learn the subject.

Finally, I thank my family back home and all my friends in Ames who have been more than my family and supported me unconditionally during my graduate education.

ABSTRACT

Statistical graphics play an important role in exploratory data analysis, model checking and diagnosis. Recent developments suggests that visual inference helps to quantify the significance of findings made from graphics. In visual inference, lineups embed the plot of the data among a set of null plots, and engage a human observer to select the plot that is most different from the rest. If the data plot is selected it corresponds to the rejection of a null hypothesis. With high dimensional data, statistical graphics are obtained by plotting low-dimensional projections, for example, in classification tasks projection pursuit is used to find low-dimensional projections that reveal differences between labelled groups. In many contemporary data sets the number of observations is relatively small compared to the number of variables, which is known as a high dimension low sample size (HDLSS) problem. The research conducted and described in this thesis explores the use of visual inference on understanding low dimensional pictures of HDLSS data. This approach may be helpful to broaden the understanding of issues related to HDLSS data in the data analysis community. Methods are illustrated using data from a published paper, which erroneously found real separation in microarray data. The thesis also describes metrics developed to assist the use of lineups for making inferential statements. Metrics measure the quality of the lineup, and help to understand what people see in the data plots. The null plots represent a finite sample from a null distribution, and the selected sample potentially affects the ease or difficulty of a lineup. Distance metrics are designed to describe how close the true data plot is to the null plots, and how close the null plots are to each other. The distribution of the distance metrics is studied to learn how well this matches to what people detect in the plots, the effect of null generating mechanism and plot choices for particular tasks. The analysis was conducted on data collected from Amazon Turk studies conducted with lineups for studying an array of exploratory data analysis tasks. Finally an R package is constructed to provide open source tools to use visual inference and distance metrics.

CHAPTER 1. INTRODUCTION

1.1 Background

Plotting data has its origins long before the development of the classical inference procedures, and then developed alongside these methods. The first recorded instance of statistical graphics based on the data was known to be in the year 1644 (variations in determination of longitude between Toledo and Rome as illustrated by Friendly and Denis (2001)). The development of inferential procedures started with Bernoulli (1700s) and gathered speed with Fisher (early 1900s) and has continued strongly through to present times (Hald, 2004).

The importance of statistical plots in statistical data analysis is widely understood. Model diagnosis and exploratory data analysis is predominantly dependent on statistical plots. Cleveland and McGill (1984) began to formalize development of graphical methods with experiments in visual perception. Wickham (2009), building on ideas originating in Wilkinson (1999), developed and implemented a grammar of graphics which presents a structured way to generate specific graphics from data and helps to define connections between disparate types of plot. Statistical graphs has been widely used for going beyond the standard paradigms of estimation and testing, to look for patterns in data beyond the expected. As pointed out by Gelman (2004), improvements in technology has helped in the development of statistical graphics. Higher resolution graphics, more sophisticated user interfaces and accessible software such as R (R Core Team, 2013) has made graphical methods to be more widely available. The problem is, although we can explore and represent our findings using statistical graphics, it has been difficult to say that what we see is “real”. This thesis research helps to fill this void.

1.2 Review of Hypothesis Testing

Classical statistical inference can be broadly classified into two categories, namely estimation and testing of hypothesis. In testing of hypothesis we start out with a claim or belief about the population parameter. We need to verify the claim or belief based on whether our sample data matches the belief. In any test, there are two competing hypotheses. The null hypothesis denoted by H_0 is a statement of what we assume to be true which reflects the current condition about the population parameter. On the other hand, the alternative hypothesis, denoted by H_a which is a statement against the null hypothesis H_0 is what we want to show.

The philosophy behind a statistical hypothesis is the same as in a jury trial. There are only two possibilities:

- “not guilty” corresponding to H_0
- “guilty” corresponding to H_a

Like in a jury trial the philosophy is “innocent until proven guilty”, we assume H_0 is true until we have sufficient evidence in the data in favor of H_a . We may have three different types of alternative hypotheses against the null hypothesis. Let us assume we want to test for a population mean μ . So against the null hypothesis $H_0 : \mu = \mu_0$, we may have three choices of alternatives:

- $H_a : \mu > \mu_0$
- $H_a : \mu < \mu_0$
- $H_a : \mu \neq \mu_0$

where μ_0 is some pre-specified value that we assume holds true under H_0 . The first two alternative hypotheses are known as one-sided alternatives and the third one is known as two-sided alternative. Also H_0 and H_a should always contradict each other, and jointly cover the population parameter space.

Let us assume that we want to test

$$H_0 : \mu = \mu_0 \quad \text{vs} \quad H_a : \mu > \mu_0$$

Now based on the sample we have in hand, we calculate the appropriate sample statistic. So in this case we calculate the sample mean \bar{x} and standard deviation s from sample of size n . If H_a is indeed true, we should expect \bar{x} to be greater than μ_0 . Then the question of interest is how much greater than μ_0 should \bar{x} be before we start doubting the null hypothesis. In other words, is the value of \bar{x} unusually large if it is really true that $H_0 : \mu = \mu_0$. If the answer is yes, then that would be evidence against H_0 in favor of H_a . To assess how unusual or unlikely our value of \bar{x} is we need to know something about how the statistic, \bar{X} , might vary from one sample to another if H_0 were really true. (Kutner et al. (2005) provides extensive explanations of these ideas.)

Assuming that the sample comes from a normal population or the sample size is large enough so that the sampling distribution of \bar{X} is approximately normal, the standardized score or the test statistic under H_0 , also known as the t -score is given by

$$t = \frac{\bar{X} - \mu_0}{S/\sqrt{n}}$$

Under H_0 , t follows a t_{n-1} distribution. From this model we can determine the probability of observing that particular value of t or something bigger, which is called the p -value. (See, for example, Moore et al. (2009).) If it is small then it is pretty unlikely, which is evidence against H_0 , which would lead us to believe that the sample comes from a distribution where $\mu > \mu_0$, that is, H_a . This is considered to be rejecting the null hypothesis.

More generally we can write the test statistic for any population parameter as

$$\frac{\text{sample estimate of parameter} - \text{hypothesized parameter value under } H_0}{\text{standard error of the estimator}}$$

Under different situations, we would hope to be able to determine the distribution of this test statistic in order to compute the p -value.

The next step is the definition of how small is small. This is determined by the level of significance, α , also called the Type I error. It is the controlled error, the probability that we are wrong in rejecting H_0 when H_0 is really true. The value of α is set to a level that we are willing to risk being wrong, typically 0.05, but sometimes 0.1 or 0.01, or even lower. Deciding whether the p -value is small corresponds comparing it with α :

- Reject H_0 if $p\text{-value} < \alpha$
- Fail to reject H_0 if $p\text{-value} > \alpha$

Equivalently we can also decide to reject or fail to reject H_0 by first determining the $100(1 - \alpha)$ percentile value of t_{n-1} distribution, called the critical value, $t_{n-1}(\alpha)$. This is compared to the observed value of the test statistic, t , leading to the decision criteria (Figure 1.1) being:

- Reject H_0 if $t > t_{n-1}(\alpha)$
- Fail to reject H_0 if $t < t_{n-1}(\alpha)$

Different alternative hypothesis require slightly different comparisons. The two-sided alternative, $H_a : \mu \neq \mu_0$, requires using:

- Reject H_0 if $|t| > t_{n-1}(\alpha)$
- Fail to reject H_0 if $|t| < t_{n-1}(\alpha)$

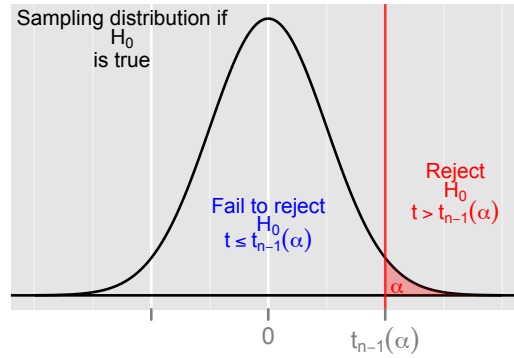


Figure 1.1 Decision regions for classical inference for $H_0 : \mu = \mu_0$ vs $H_a : \mu > \mu_0$.

Type I error, is the probability of rejecting the null hypothesis H_0 when H_0 is true. Type II error, denoted by β is the probability of failing to reject the null hypothesis H_0 when H_0 is false. Type I error is committed in a jury trial when it is decided that a not guilty person is “guilty”. This is a serious mistake as an innocent person is punished. Type II error is committed when there is

a guilty person is not convicted, not considered to be so serious. The power of a statistical test is defined as the probability that the test will reject H_0 when H_0 is false i.e the power of the test is the probability of correctly rejecting a false null hypothesis. So the power is the probability of not committing a Type II error and hence is denoted by $1 - \beta$. (Casella and Berger (2002) and Lehmann (1997) give more thorough treatments of hypothesis testing.)

1.3 Introduction to Visual Inference

Buja et al. (2009) proposes visual statistical methods with an inferential framework. In visual inference the plots take on the role of test statistics, the test statistic is a visual representation of the data, not a numerical value. Comparison data is generated under the assumption that the null hypothesis is true, and plots of this data are generated. These plots, known as the null plots gives the “null distribution of plots” analogous to the null distribution of test statistics. The plot of the data is compared with the null plots. Variations of these ideas have historically been utilized for data analysis, albeit sparingly, which is commented in the introduction of Buja et al. (2009). Gelman (2004) puts these ideas in the context of model building. The key feature of Buja et al. (2009) is that it makes the connection to the process of hypothesis testing, and quantifying significance. There are two protocols defined in this paper.

1.3.1 Protocols of Visual Inference

Buja et al. (2009) introduces two protocols for graphical inference: one is the “Rorschach” and the other is the “lineup”. The purpose of the Rorschach protocol is to measure a data analyst’s tendency to over interpret plots in which there is no or spurious structure. On the other hand the lineup provides a simple inferential process to produce a valid p-value for the observed plot. Here we describe the protocols briefly and refer the reader to Buja et al. (2009) and Wickham et al. (2010) for more details.

- **Rorschach:** It is possible that the randomness of the data inherits some pattern in the plot.

The Rorschach protocol is designed to expose the data analyst’s tendency of over-interpretation

of patterns when there is actually no or spurious structure. The results are specific to a particular data analyst and a particular data analysis procedure. The protocol estimates the effective family-wise Type I error rate. A data administrator may generate the null plots and decides about the prior information that the data analyst is provided. The administrator may program the series of null plots in such a way that the plot of the real data is inserted in a random location. A toned-down version may also be used for self training. This self training may improve the family-wise error rate of the data analyst and develop an awareness of the features they are most likely to spuriously detect. The Rorschach protocol is named after the (pop-)psychology Rorschach test, in which subjects interpret abstract ink blots.

- **Lineup:** The lineup protocol gets its name after the police lineup of criminal investigation. In a police lineup, the accused is placed among a set of innocent people who may be prisoners, actors or volunteers having no connection with the case. The witness is asked to pick from this lineup. Likewise in a lineup protocol, the accused which is the observed plot is placed randomly among a set of null plots, say m , and the witness (in this case the viewer) is asked to identify the plot as most different from the others. If the viewer can correctly identify the observed plot from the lineup, we have reasons to believe that the observed plot has a specific pattern which is missing in the null plots. This protocol leads to the development of the technique of visual inference by defining the test statistic as a plot that mostly show a specific pattern in the data when alternative hypothesis is true. Figure 2.2 shows a typical lineup.

Let us consider the following example. The data represents the concentration of a metal in mg/kg for two sites A and B. We want to test whether there exists a significant difference between the concentration levels in the two sites A and B. Let μ_1 denote the mean concentration level in Site A and μ_2 denote the mean concentration level in Site B. To test that, we have the following null and alternative hypothesis:

$$H_0 : \mu_1 = \mu_2 \quad \text{versus} \quad H_a : \mu_1 \neq \mu_2$$

(Technically the problem this data addressed was more interested in testing a one-sided alternative, whether site B has higher concentration than site A, but it is more interesting for this example to

consider the two-sided alternative hypothesis.) The test statistic is the plot of the real data. The 19 null plots are generated by assuming that null hypothesis $H_0 : \mu_1 = \mu_2$ is true. So we permute the class variable site to obtain the null plots keeping the other variables fixed. The observed plot is placed randomly among these 19 plots in a lineup given in Figure 2.2. The viewer is asked to identify the plot which is most different. If the viewer can identify the plot of the real data, we will have reasons to believe that the observed plot has a pattern which is absent in the null plots. So we would reject the null hypothesis. If the viewer cannot identify the observed plot, we fail to reject the null hypothesis.

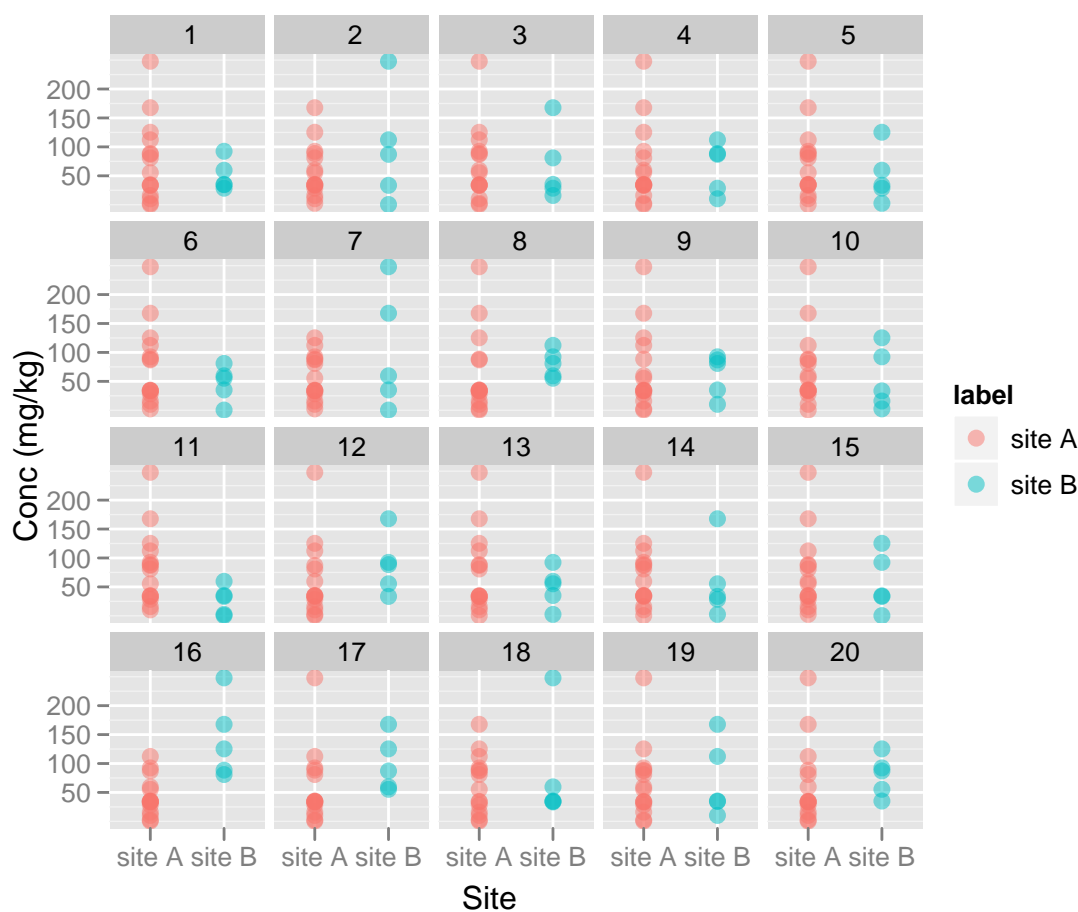


Figure 1.2 A typical lineup plot ($m = 20$) for testing $H_0 : \mu_1 = \mu_2$. When the alternative hypothesis is true the observed plot should have the largest vertical difference between the centers. Can you identify the observed plot?

Plot 16 is the plot of the real data. If the viewer could identify the plot then we have reasons to believe that there exists a statistically significant difference between the mean concentration levels in site A and site B. So the lineup protocol is the basis of the visual inference while the Rorschach protocol helps viewer understand the extent of randomness.

Majumder et al. (2013) describes a comparative study between the visual inference method and the classical inference methods, focusing on plots that might be used in linear modeling. In his work the expected power of the visual test is compared with the power of the uniformly most powerful (UMP) test. The power of the visual test is computed by responses from several large samples of lineup evaluators recruited through Amazon Turk (Amazon, 2010). The results suggest that the expected power of a visual test is almost as good as the power of UMP test, that visual inference compares favorably with classical testing, in the traditional setting where the classical test performs well. They established properties and efficacy of visual testing procedures in order to use them in situations where traditional test cannot be used. In addition Majumder et al. (2013) provide a nice way of making the leap traditional hypothesis testing to visual inference. We have adapted that table for the $H_0 : \mu_1 = \mu_2$ vs $H_a : \mu_1 \neq \mu_2$ example described and plotted in Figure 2.2, which can be seen in Table 2.1.

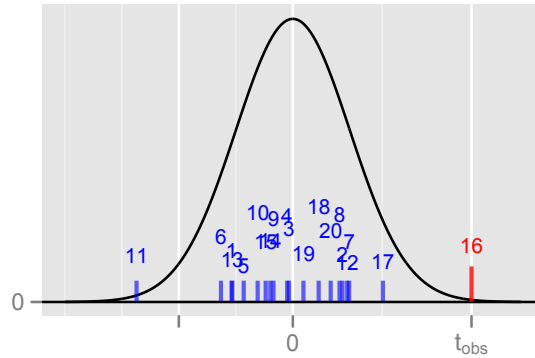
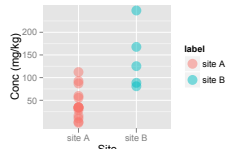
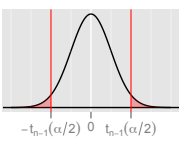
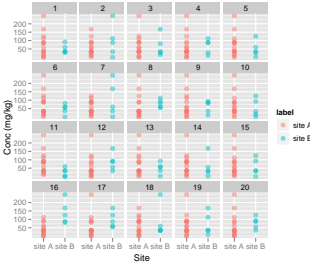


Figure 1.3 Sampling distribution of the test statistic with the observed value and the values for the null plots corresponding to the lineup in Figure 2.2.

In traditional hypothesis testing the sampling distribution of the test statistics is continuous, which allows evaluation of probability on an infinite spectrum. With the lineup, although conceptually

Table 1.1 Comparison of visual inference with traditional hypothesis testing.		
	Mathematical Inference	Visual Inference
Hypothesis	$H_0 : \mu_1 = \mu_2$ vs $H_a : \mu_1 \neq \mu_2$	$H_0 : \mu_1 = \mu_2$ vs $H_a : \mu_1 \neq \mu_2$
	↓	↓
Test Statistic	$T(y) = \frac{\bar{y}_1 - \bar{y}_2}{s \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$	$T(y) =$ 
	↓	↓
Sampling Distribution	$f_{T(y)}(t);$ 	$f_{T(y)}(t);$ 
	↓	↓
Reject H_0 if	observed T is extreme	observed plot is identifiable

we may have an infinite collection of plots from the null distribution, in practice, we sample a finite number of null datasets to generate the lineup. A human judge has a physical limit on the number of null plots they can peruse. This poses one of the issues with using the lineup protocol. Figure 1.3 gives the sampling distribution (black curve) for the t -distribution, along with the t -statistics of the samples that were drawn from the null distribution (blue bars) and that of the observed data (red bar) for the plots in the lineup shown in Figure 2.2. Effectively, in visual inference the red line is compared only to these finite number of blue lines visually to make a decision, unlike classical inference where we look at the rejection region (Figure 1.1) to make decisions. So as Tukey suggested, there may be a “bad” random sample of null plots which may affect our decision. This is a major component of this thesis research to we develop techniques to determine the quality of a lineup. In practice, though, it needs to be noted, that visual inference will not typically be used in applications where there is an existing classical test. The purpose of visual inference is not to compete with classical statistics – its purpose is to provide formalism and quantification in problems where there are none, currently. For the purposes of research and assessment we use the classical setting because it provides benchmarks for how visual inference will likely perform.

1.4 Overview

My research extends the visual inference methodology in two ways – by producing metrics to quantify lineups and examine how people read statistical plots, and examining how it applies to assessing dimension reduction in high dimension low sample size (HDLSS) problems.

Chapter 2 explores the performance of dimension reduction methods such as projection pursuit for high dimension, low sample size (HDLSS) data. The key points of interest were producing a way to evaluate an algorithm when its results are primarily visual, and to examine how well people can distinguish between real separation and noise in HDLSS data. Several human subjects experiment were conducted using simulated data to provide controlled conditions. Results suggest that people can detect real separation from pure noise up to a reasonably high dimension. The broader community still has difficulty in understanding HDLSS data, which can be seen by published

papers where the authors get excited about structure in low-dimensional projections, when it is really present simply because of the sparseness of high-dimensions. This chapter suggests that the lineup protocol can potentially explain HDLSS issues effectively. This paper is accepted for publication by Computational Statistics.

Chapter 3 focusses on developing metrics to describe the quality of a lineup. In conventional inference, the test statistic is compared to all possible values of the sampling distribution, but in visual inference only a finite number of plots are drawn randomly from the sampling distribution. Understanding the effect of these finite comparisons is important. This chapter develops a variety of distance metrics that can be used to measure the “closeness” of observed data plot with the null plots. These metrics are compared to the results from human subjects experiments. These metrics may be useful to learn how people detect the observed data plot from the null plots. For example, with the HDLSS simulations, traditional class separation metrics like WBratio do not match the results from people as well as a metric measuring the gap between clusters.

Chapter 4 describes the R package **nullabor** which generates the lineup plots automatically if the observed data and the null generating mechanism are provided. Routines to calculate the distance metrics on the lineup were added. Some common distance metrics are included in the package and users have the freedom of using their own distance metrics. The package also provides diagnostic plots for comparing a lineup with other possible lineups that may have been generated.

Finally, chapter 5 provides a summary of the dissertation and discussion of possible future research work.

1.5 Scope of my Research

This dissertation provides the ground work for the application of visual inference. It applies visual inference methods in a high dimension, low sample size (HDLSS) framework, including using

it to show that a gene expression data set does not have the separate clusters as claimed. The research has initiated ideas on metrics that could be used to evaluate the effect of the finite set of null plots in a lineup, cross-validate observer data, and help understand observer responses. Finally, an open source package has been extended to include this new metric methods to improve the use of the lineup protocol for visual inference.

CHAPTER 2. USING VISUAL STATISTICAL INFERENCE TO BETTER UNDERSTAND RANDOM CLASS SEPARATIONS IN HIGH DIMENSION, LOW SAMPLE SIZE DATA

A paper accepted by *Computational Statistics*.

Niladri Roy Chowdhury, Dianne Cook, Heike Hofmann, Mahbubul Majumder

Eun-Kyung Lee, Amy L. Toth

Abstract

Statistical graphics play an important role in exploratory data analysis, model checking and diagnosis. With high dimensional data, this often means plotting low-dimensional projections, for example, in classification tasks projection pursuit is used to find low-dimensional projections that reveal differences between labelled groups. In many contemporary data sets the number of observations is relatively small compared to the number of variables, which is known as a high dimension low sample size (HDLSS) problem. This paper explores the use of visual inference on understanding low-dimensional pictures of HDLSS data. Visual inference helps to quantify the significance of findings made from graphics. This approach may be helpful to broaden the understanding of issues related to HDLSS data in the data analysis community. Methods are illustrated using data from a published paper, which erroneously found real separation in microarray data, and with a simulation study conducted using Amazon's Mechanical Turk.

keywords : statistical graphics , lineup, visualization , projection pursuit, data mining

2.1 Introduction

Many problems needing solutions today require the analysis of data where more variables are measured than there are samples taken. This is commonly referred to as high dimensional, low sample size (HDLSS) data (see for e.g. Hall et al. (2005)). HDLSS data occur in many application areas like face recognition, spectroscopy and gene expression analysis. Classical statistical methods often fail in this context, because of insufficient data to support for parameter estimation.

Reducing the dimension, using principal component analysis (PCA), would be a classical first step in the analysis of HDLSS data. PCA requires estimating the eigenvalues (maximum variance) and eigenvectors (direction of maximum variance) of the population variance-covariance based on the sample. With insufficient data this is a Sisyphean task. Just imagine, estimating a line on the foundation of a single point - there are infinitely many possibilities for lines. For classification tasks, finding a low-dimensional space where the classes are separated is a common first step. Linear discriminant analysis (LDA) is the classical approach. LDA solves an eigenvalue decomposition problem comparing distances between group means with variance around each mean. Estimating the variance-covariance is problematic when there are few points. In addition, when there are few sample points in high dimensions, differences between groups can be found in many different low-dimensional spaces, simply because of the sparseness of space.

Marron et al. (2007) describes the estimation issues associated with HDLSS. Advancements in PCA to handle HDLSS data have been done by Jung et al. (2012) and Yata and Aoshima (2011). Donoho and Jin (2009) and Donoho and Jin (2008) study optimal variable selection and introduce a principle of model selection for problems where only a small fraction of the variables are useful and unknown. Penalization is another common approach to handle HDLSS, and has been applied to classification problems e.g. (Witten and Tibshirani, 2011; Lee and Cook, 2010). Estimates of the variance-covariance are obtained by an interpolation with the identity matrix, effectively reducing the importance of some variables.

So, although substantial research has produced many new approaches to creating better models and estimation for HDLSS data, the major issues are still not clear to many data analysts. For

example, Toth et al. (2010) make a common mistake of seeing structure where none exists. Figure 2.1 reproduces the result in this paper. They use LDA to examine gene expression data of wasps containing 447 variables and 50 cases. There are 50 different paper wasps divided into 4 types: Foundress (F), Gyne (G), Queen (Q) and Worker (W), 14 wasps of type Foundress and 12 each of the other 3 types. The authors, knowing that LDA requires that the dimension (p) should be smaller than the number of observations (n), first reduced the dimension from 447 to 40 by randomly selecting a subset of significantly different oligonucleotides. LDA produced a 2D projection ($d = 2$) of best separation. This is almost the same approach as used in Dudoit et al. (2002), one of the first studies of classification of gene expression data. What results is a picture of the four groups that suggests big differences in the types of wasps. There exists no conventional inferential method that enables us to conclude whether this apparently clear separation is statistically significant or not. For prediction, typically data is broken into training and test sets, or cross-validation is conducted to assess the significance of difference, using test set error. This approach does not work well for visualization.

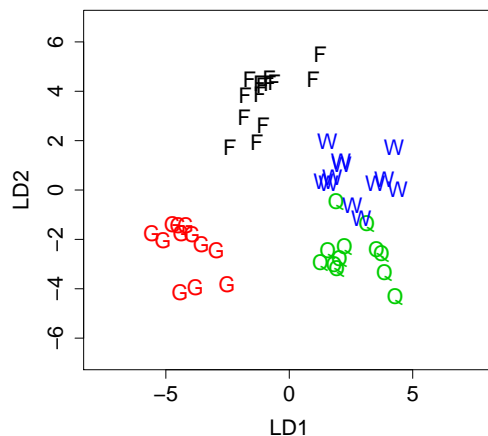


Figure 2.1 LD1 versus LD2 from an LDA on a randomly selected subset of 40 significantly different oligos : F, Foundress; G, gyne; Q, queen and W, worker. It can be noticed that the groups F and G are separated. This plot is generated to match Figure 2 in Toth et al. (2010).

We propose that new methods for inference on graphics might be helpful for building under-

standing very generally. Visual statistical inference was first conceptually introduced by Buja et al. (2009), formalized and validated by Majumder et al. (2013). Using visual inference, it can be shown that there is no real difference between the wasp groups - what you see is a mirage.

Visual inference may also be useful in related applications, such as checking algorithms that produce visual results. We used the approach described in this paper to check the optimization algorithm of projection pursuit in the `tourr` package (Wickham et al., 2011) in R (R Core Team, 2013). The optimization procedure was new, and we suspected that it was being sensitive to the order of data values, and returning projections of purely noise data that we thought were surprisingly distinct from noise. Visual inference was able to temper our concerns.

This paper describes visual statistical inference as applied to dimension reduction for HDLSS. In particular we focus on dimension reduction using projection pursuit, and the effect that having high dimension has on the robustness of the separation between groups. Small simulation experiments are used to examine the problem in a controlled setting. The next section explains visual inference methods. Section 2.3 discusses the dimension reduction methods. Section 2.4 describes Amazon’s Mechanical Turk (Amazon, 2010) which was used to conduct the experiment. The application of visual inference methods on the wasp data (Toth et al. (2010)) is described in Section 2.5. Section 2.6 discusses the experiment designed to examine people’s perception of separation in the presence of real separation and “purely noise” for simulated HDLSS data. Section 3.8 discusses the collected data and results.

2.2 Visual inference methods

Buja et al. (2009) proposed two protocols, the Rorschach and the lineup. While the Rorschach protocol helps to understand the extent of randomness, the lineup protocol is used for testing significance of findings. These methods together are called visual statistical inference. Majumder et al. (2013) made a head-to-head comparison between visual statistical inference tests and classical tests which showed that the lineup protocol performs similarly to the classical tests. Unlike classical hypothesis testing, the test statistic in visual inference is not numeric, but a plot that is appropriately

chosen to display a distinctive pattern in case that the null hypothesis is false. The lineup protocol of size m embeds the observed data plot amongst $(m - 1)$ null plots. Null plots are created by a mechanism consistent with the null hypothesis. Human subjects are asked to identify the plot in the lineup with the most distinct feature(s). When the alternative hypothesis is true, it is expected that the plot of the observed data, the test statistic, will have visible feature(s) inconsistent with the null hypothesis. If the subjects choose the plot of the observed data, this is evidence against the null hypothesis and with enough support, the null hypothesis is rejected.

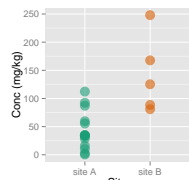
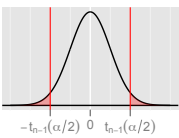
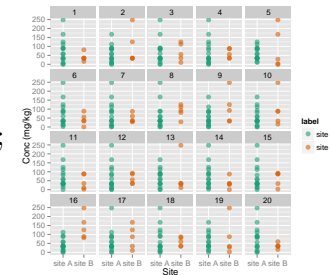
An illustration of the lineup protocol in contrast to the conventional test is shown in Table 2.1. Both start with the same hypothesis but the test statistic in a conventional setting is the parameter estimate divided by its standard error. In visual inference the test statistic is a plot of the observed data. In this case, a dot plot is used since the variable of interest is continuous with two groups. In a conventional test the value of the test statistic is compared with all possible values of the sampling distribution, the distribution of the test statistic if the null hypothesis is true. If it is extreme, then the null hypothesis is rejected. In visual inference, the plot of the data is compared with a finite number of samples drawn from the sampling distribution. If the actual data plot is selected as the most different, then the null hypothesis is rejected.

For example, suppose we have two sample data on the concentration of a metal in mg/kg for sites A and B of sizes n_1 and n_2 respectively. We want to test whether there exists a difference between the concentration levels in the two sites A and B. To test for statistically significant difference between the two populations from which the data was sampled, let μ_1 denote the mean concentration level in Site A and μ_2 denote the mean concentration level in Site B. Thus the null and alternative hypothesis would be

$$H_o : \mu_1 = \mu_2 \quad \text{versus} \quad H_a : \mu_1 \neq \mu_2$$

The conventional test would be a two-sample t-test with test statistic $T(y)$ described in Table 2.1. One way to plot this data is a side-by-side dotplot. Let this be the visual test statistic $V(y)$. Null plots are generated assuming that H_o is true. Here, this is achieved by randomly permuting the site label. The observed data plot is placed randomly among the null plots to obtain a lineup.

Table 2.1 Comparison of visual inference with traditional hypothesis testing. Starting with the same hypothesis, the test statistic in a conventional setting is a real number while in visual inference it is a plot of the observed data. In conventional testing the value of the test statistic is compared with all possible values of the sampling distribution. H_o is rejected if it is extreme. In visual inference, the plot of the data is compared with a finite number of samples drawn from the null distribution. If the actual data plot is identifiable, then the null hypothesis is rejected.

	Mathematical Inference	Visual Inference
Hypothesis	$H_o : \mu_1 = \mu_2$ vs $H_a : \mu_1 \neq \mu_2$	$H_o : \mu_1 = \mu_2$ vs $H_a : \mu_1 \neq \mu_2$
	↓	↓
Test Statistic	$T(y) = \frac{\bar{y}_1 - \bar{y}_2}{s \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$	$V(y) =$ 
	↓	↓
Sampling Distribution	$f_{T(y)}(t);$ 	$f_{V(y)}(t);$ 
	↓	↓
Reject H_o if	observed T is extreme	observed data plot is identifiable

In this example, $m = 20$ is the size of the lineup - there are 19 null plots. If H_o is not true, the dots of one group should be vertically shifted relative to the other group. If the human observer can identify the plot of the real data, there will be reason to believe that the observed data plot has a pattern which is absent in the null plots leading to a rejection of the null hypothesis. If the viewer cannot identify the observed data plot, we fail to reject the null hypothesis. Under the null hypothesis, each observer has a $1/m$ chance of picking the observed plot from a lineup of size m . Hence $1/m$ is the minimal value at which we can set the Type I error, α , consistent with $\alpha = 0.05$ if $m = 20$. Majumder et al. (2013) provides more detailed discussion about this. For this problem, visual inference enables the handling of the small sample size and non-normality of the population. However, in general the setting for visual inference would be problems where no conventional test exists.

Majumder et al. (2013) describes the methods of obtaining the power of the visual test, by combining results from multiple users. For their simulation experiments human observers were recruited through Amazon’s Mechanical Turk (Amazon, 2010). Power of the visual test used in their simulation was also calculated theoretically. Their results suggest that the power of visual statistical inference is comparable to conventional tests in a setting of testing the parameters of linear regression models. The subject specific power of the visual test can also be estimated from the multiple responses data from each human observer, which might help quantify individual visual skills.

2.3 Dimension reduction

Projection pursuit [e.g. (?)] is used for dimension reduction in our studies. Projection pursuit (PP) finds the most interesting low dimensional projection of high dimensional data by maximizing some criterion of interest, e.g. variance or clustering or group separation. As pointed out in Huber (1985) the most exciting feature of projection pursuit is that it can bypass the curse of dimensionality.

In classification problems, linear discriminant analysis (LDA) can be used to find a low-dimensional space where the groups are most separated. This corresponds to using the LDA

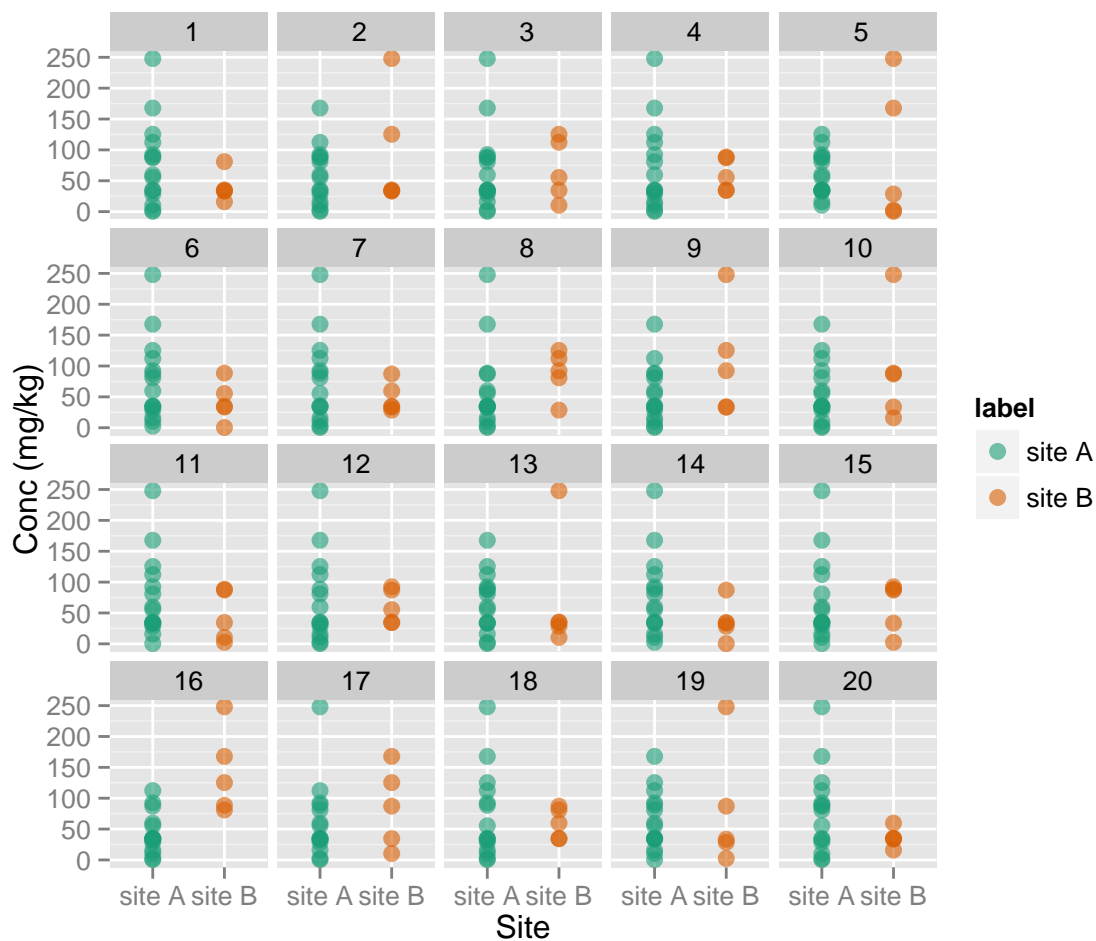


Figure 2.2 A typical lineup ($m = 20$) for testing $H_o : \mu_1 = \mu_2$. When the alternative hypothesis is true, the observed data plot should have the largest vertical difference between the centers. Can you identify the observed data plot? The solution to the lineup is provided in the Appendix.

index (Lee and Cook, 2010) in projection pursuit. Let \mathbf{X}_{ij} be the p -dimensional vector of the j th observation in the i th class, $i = 1, \dots, g, j = 1, \dots, n_i$, g is the number of classes, n_i is the number of observations in class i , and $n = \sum_{i=1}^g n_i$. Let $\bar{\mathbf{X}}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \mathbf{X}_{ij}$ be the i th class mean and $\bar{\mathbf{X}}_{..} = \frac{1}{n} \sum_{i=1}^g \sum_{j=1}^{n_i} \mathbf{X}_{ij}$ be the total mean. The LDA PP index is

$$I_{LDA}(\mathbf{A}) = \begin{cases} 1 - \frac{|\mathbf{A}^T \mathbf{W} \mathbf{A}|}{|\mathbf{A}^T (\mathbf{W} + \mathbf{B}) \mathbf{A}|} & \text{for } |\mathbf{A}^T (\mathbf{W} + \mathbf{B}) \mathbf{A}| \neq 0 \\ 0 & \text{for } |\mathbf{A}^T (\mathbf{W} + \mathbf{B}) \mathbf{A}| = 0 \end{cases} \quad (2.1)$$

where \mathbf{A} is an orthogonal projection onto a k -dimensional space and

$$\begin{aligned} \mathbf{B} &= \sum_{i=1}^g n_i (\bar{\mathbf{X}}_i - \bar{\mathbf{X}}_{..})(\bar{\mathbf{X}}_i - \bar{\mathbf{X}}_{..})^T : \text{between-class sums of squares,} \\ \mathbf{W} &= \sum_{i=1}^g \sum_{j=1}^{n_i} (\mathbf{X}_{ij} - \bar{\mathbf{X}}_i)(\mathbf{X}_{ij} - \bar{\mathbf{X}}_i)^T : \text{within-class sums of squares.} \end{aligned}$$

For HDLSS data, the penalized discriminant analysis (PDA) index (Lee and Cook, 2010) is more robust. Let \mathbf{X}_{ij}^* be the standardized vector of \mathbf{X}_{ij} . Then

$$\begin{aligned} \mathbf{B}^s &= \sum_{i=1}^g n_i (\bar{\mathbf{X}}_i^* - \bar{\mathbf{X}}_{..}^*)(\bar{\mathbf{X}}_i^* - \bar{\mathbf{X}}_{..}^*)^T : \text{between-class sums of squares of the standardized data} \\ \mathbf{W}^s &= \sum_{i=1}^g \sum_{j=1}^{n_i} (\mathbf{X}_{ij}^* - \bar{\mathbf{X}}_i^*)(\mathbf{X}_{ij}^* - \bar{\mathbf{X}}_i^*)^T : \text{within-class sums of squares of the standardized data} \end{aligned}$$

where $\bar{\mathbf{X}}_i^*$ is the i th class mean of the standardized data and $\bar{\mathbf{X}}_{..}^*$ is the total mean of the standardized data, which is 0. The PDA index is defined as

$$I_{PDA}(\mathbf{A}, \lambda) = 1 - \frac{|\mathbf{A}^T \{(1 - \lambda) \mathbf{W}^s + n \lambda \mathbf{I}_p\} \mathbf{A}|}{|\mathbf{A}^T \{(1 - \lambda) (\mathbf{B}^s + \mathbf{W}^s) + n \lambda \mathbf{I}_p\} \mathbf{A}|} \quad (2.2)$$

where \mathbf{A} is an orthonormal projection onto a k -dimensional space and $\lambda \in [0, 1)$ is a predetermined parameter. Penalized LDA (Witten and Tibshirani, 2011) is a similar approach.

These indices are available for projection pursuit using the `tourr` package (Wickham et al., 2011) in R (R Core Team, 2013). The `tourr` package produces tours of multivariate data. The package

also includes functions for creating different types of tours like grand, guided and little tours, which project multivariate data with p dimensions to 1, 2, 3 or d dimensions where $d \leq p$. The guided tour function is used here. The guided tour will converge to a maximally interesting projection. Here, that is a projection where groups show the biggest separation. For this paper we used $d = 1$ or 2 .

2.4 Amazon Turk experiments

Amazon’s Mechanical Turk (Amazon, 2010) is a service that enables researchers to employ people to do tasks which computers perform poorly. In exchange for their efforts, the subjects are paid, not substantially, but on the scale of the minimum wage of the USA. For visual inference studies, subjects are typically given a block of ten lineups to evaluate during a job. From this block, one lineup is typically used as a filter, and the remaining lineups produce data for the studies. Because a subject evaluates more than one lineup, and a lineup is evaluated by more than one subject, we obtain some replication in the results upon which to estimate variation. The one filter lineup, in which the observed data plot is markedly different from the nulls, is necessary because Turkers are not manually monitored, and a few attempt to maximize financial gain without taking the exercise seriously.

For this paper, two Turk studies were run, one for the wasps data, and the other for the simulation study. Turkers are redirected from Amazon to a website which describes the study in detail, provides some practice trials, collects demographic details, and responses. The website of the simulation study is http://mahbub.stat.iastate.edu/feedback_turk7/homepage.html.

2.5 Wasps application

We return to the motivating example. Figure 2.1 suggested that the expression patterns of the wasp groups are different. The question of interest is “Is this separation real?” This can be investigated by testing the hypothesis:

H_o : There is NO difference in the expression levels between the types of wasp.

H_a : At least one of the types of wasps has different expression levels.

A lineup is made of the wasp data obtained from Toth et al. (2010) to test H_o where the null plots are made by permuting the wasp type label, and re-doing the LDA. If there is a real difference between the expression levels for the types of wasps then the observed data plot should be detectable in the lineup. Figure 2.3 shows a lineup. Three different lineups were created using this procedure.

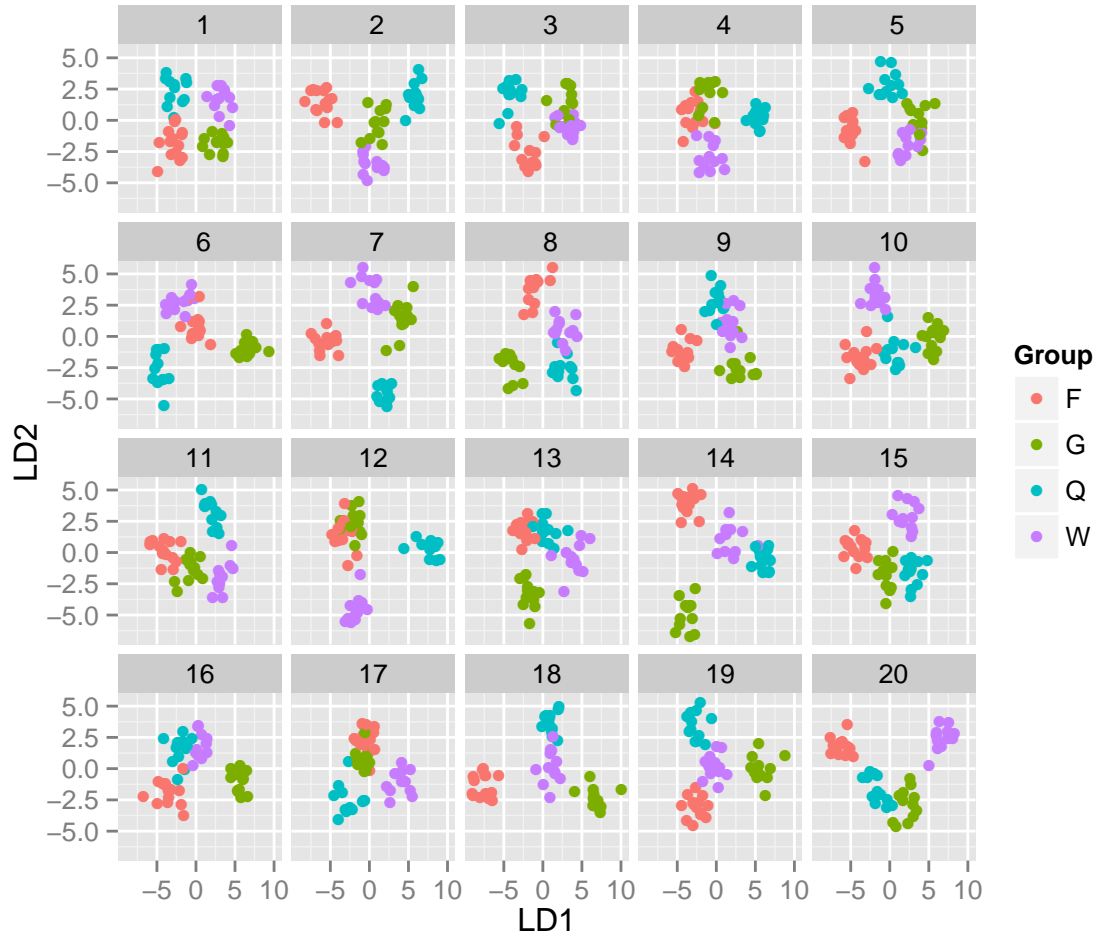


Figure 2.3 Lineup of the wasps data. One plot shows the observed data and the remaining 19 show null data where the wasp type labels were randomly assigned. Each plot was produced by conducting LDA on the 40D data to produce the 2D projection with best separation. Which plot shows the most separation between the 4 groups? The solution is provided in the Appendix.

In addition, three more lineups are made containing only null plots, with one plot randomly chosen to act as an observed data plot. These lineups were shown to the subjects recruited from Amazon Turk. A total of 116 subjects evaluated the 6 lineups. Table 2.2 shows the results. The

detection rate for the plot of the wasp data is 0! This is worse than that of purely noise data. You will notice that for one of the purely noise lineups, subjects very often detected the (random) observed data plot. This happened because the randomly generated observed data plot actually had more separation than any other plot in that lineup. This is the nature of randomness, but makes for interesting results here. The p -value is calculated according to the procedure given by Majumder et al. (2013). The large p -values indicate that there is no statistically significant evidence upon which we reject the null hypothesis. Thus we have to conclude that the separation in the wasp data (Toth et al. (2010)) is not real. It is purely the effect of high dimensionality.

Table 2.2 Results of the Turk study on the wasps data. Detection rate for each lineup is shown, with the number of subjects, and p -value associated. The detection rate is highest for one of the purely noise lineups, which occurred because the plot with the most difference between groups happened to be the one that is randomly generated as the “real” data. Averaging the p -values for each set of lineups, for the wasps is 1.0, and for the pure noise, is 0.67 suggesting that the apparent separation in the wasp data (Toth et al. (2010)) is consistent with pure noise induced by the high dimensions.

Data	Replicate	Num Subjects	Detection rate	p -value
Wasps	1	25	0.0000	1.0000
	2	13	0.0000	1.0000
	3	27	0.0000	1.0000
Purely noise	1	19	0.2632	0.0002
	2	18	0.0000	1.0000
	3	14	0.0000	1.0000

The probability of separation by chance between two groups in purely noise data, given a fixed sample size and dimension, was quantified by Ripley (1996) (Proposition 3.1). Figure 2.4 illustrates this result for 1D projections, for different p , where sample size is fixed at $n = 30$. When $p = 2$, $P(\text{separation} | n = 30, p = 2) = 0$ and it reaches 1 when $p = 25$. For data of the size of the wasps, $n = 50$ and $p = 38$, the probability of obtaining separation with only two groups is 1, so we would expect that there would certainly be separation between four groups in 2D.

In the original paper (Toth et al., 2010), the dimensionality was reduced from 389 by choosing the genes that showed the greatest separation. So the problem of high dimensionality is actually

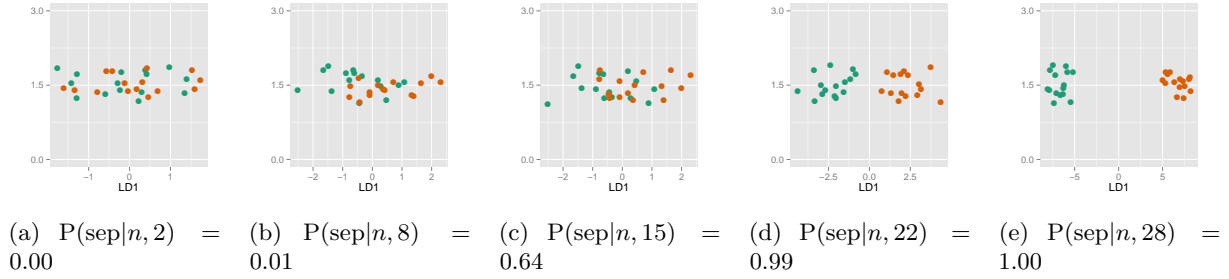


Figure 2.4 Plots showing example 1D projections for $p = 2, 8, 15, 22, 28$ and $n = 30$ for purely noise data. The probability of obtaining a projection where groups are separated is calculated and displayed below the plots. Of course, once a projection is computed the groups are either separated or not - the event occurred or didn't - and we can see that the last two plots display separated groups. The difference between the groups increases as p increases, and the likelihood of obtaining a projection with separation increases.

even worse for these data. In general, reducing the data dimensions so that the sample size is bigger than dimension is not, on its own, sufficient. It is important, even, with so few cases to do cross-validation, or break the sample into training and test sets before conducting analysis. LDA is known also to be a problem for HDLSS data, because it requires estimating more parameters than the available data allows. A better prospect for dimension reduction is the penalized discriminant analysis (PDA) index (Lee and Cook, 2010), which helps adjust for the over-estimation. Other results and the overall conclusions in Toth et al. (2010) are not affected by the inadequacy revealed by this visual inference analysis. A similar LDA performed on wasp gene expression data with a much higher sample size in Toth et al. (2007) did not suffer from the HDLSS problem. We determined that there were robust separations between the groups based on those data (results not shown).

2.6 Follow-up simulation experiment

2.6.1 Experimental design

The goal is to determine how well people can detect the presence of real separation as distinguishable from random noise. To achieve this, the experiment is set up with several factors: real separation or pure noise, data dimension and projection dimension. Real separation is achieved by

setting 1 or 2 variables with real separation among a number of noise variables. Sample size is fixed to keep the experiment manageable. Also mean difference is kept fixed. The levels of the factors used in the experiment are given in Table 2.3. Three replicates at each level are generated. These produced 60 different “observed data sets”, and thus, 60 different lineups.

Table 2.3 Levels of the factors used for the simulation experiment.

n	projection(d)	separation	dimension (p)	replicates
30	1	Yes	20, 40, 60, 80, 100	3
		No	20, 40, 60, 80, 100	3
30	2	Yes	20, 40, 60, 80, 100	3
		No	20, 40, 60, 80, 100	3

2.6.2 Simulation process

Two groups of p dimensions of data with 15 observations in each group are generated from $N(0, 1)$. The data from the first group is labeled as group 1 and the data from the second group as group 2, yielding a $30 \times (p + 1)$ matrix, \mathbf{X} , where the first 15 observations are from group 1 and the last 15 observations are from group 2. The data for both the groups is purely noise, having no dependence on the groups. So \mathbf{X} can be written as

$$\mathbf{X}^{n \times (p+1)} = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_p, \text{Group})$$

where each \mathbf{X}_i is a vector of dimension 30 for $i = 1, \dots, p$. This matrix \mathbf{X} excluding the Group variable gives the p -dimensional pure noise data or data with no separation.

To introduce real separation in the data, values for p -th variable \mathbf{X}_p are shifted apart by 6 units between the two groups:

$$\mathbf{X}_p = \begin{cases} \mathbf{X}_p - 3 & \text{if } \mathbf{X}_p \in \text{group 1} \\ \mathbf{X}_p + 3 & \text{if } \mathbf{X}_p \in \text{group 2} \end{cases}$$

and then standardized to have unit variance again. On each dataset of p dimension, a projection pursuit optimization with the PDA index is performed to obtain the 1D projection of best separation, yielding $\mathbf{Y} = \mathbf{XA}$.

The above procedure is effectively the same for 2D projections with few key differences. The first 10 observations are labelled group 1, the second 10 observations as group 2 and the last 10 as group 3. So, collectively, \mathbf{X} can be written as

$$\mathbf{X}^{n \times (p+1)} = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_p, \text{Group})$$

where each \mathbf{X}_i is a vector of dimension 30 for $i = 1, \dots, p$. This matrix \mathbf{X} excluding the Group variable gives the p -dimensional noise data or data with no separation with 3 groups.

To introduce real separation, the means of the 3 groups are adjusted in the last two dimensions i.e. \mathbf{X}_{p-1} and \mathbf{X}_p . The adjustment is done in the following way:

$$(\mathbf{X}_{p-1}, \mathbf{X}_p) = \begin{cases} (\mathbf{X}_{p-1} - 3, \mathbf{X}_p) & \text{if } (\mathbf{X}_{p-1}, \mathbf{X}_p) \in \text{group 1} \\ (\mathbf{X}_{p-1} + 3, \mathbf{X}_p) & \text{if } (\mathbf{X}_{p-1}, \mathbf{X}_p) \in \text{group 2} \\ (\mathbf{X}_{p-1}, \mathbf{X}_p + \sqrt{27}) & \text{if } (\mathbf{X}_{p-1}, \mathbf{X}_p) \in \text{group 3} \end{cases}$$

and standardized. If \mathbf{X}_{p-1} versus \mathbf{X}_p are plotted in a scatterplot, the points cluster along the vertices of an equilateral triangle of side 6. Hence the data with 2 dimensions of real separation divided into 3 groups is obtained. A projection pursuit with a PDA index is performed to obtain the 2D projection of best separation, yielding $\mathbf{Y} = \mathbf{XA}$.

2.6.3 Producing lineups

Two different visual test statistics, $V_1(\mathbf{Y})$ and $V_2(\mathbf{Y})$ are used in this paper, for representing 1D and 2D data. $V_1(\mathbf{Y})$ is a horizontal jittered dot plot, with color representing groups. $V_2(\mathbf{Y})$ is a scatterplot with color representing groups. Symbols are kept constant for uniformity of appearance. Figure 2.5 shows the two different visual test statistics.

To obtain the null plots in a lineup, the group variable is permuted in order to break any dependence between the group variable and the other variables. Projections are obtained and plotted in the same way as the test statistic. The test statistic, which is the observed data plot, is

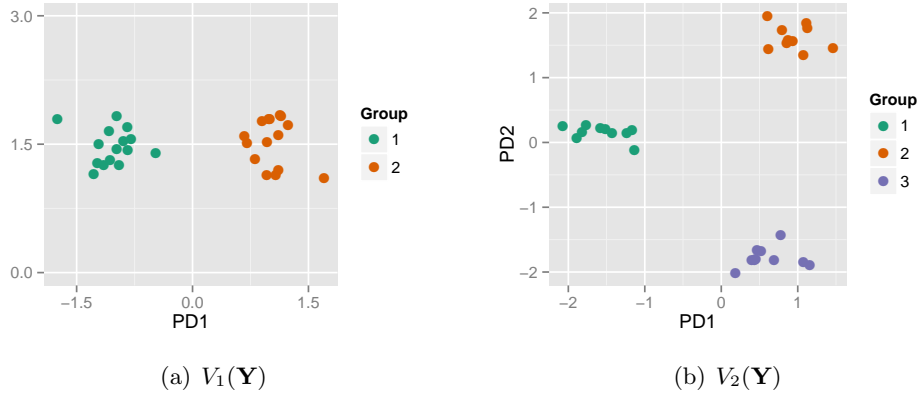


Figure 2.5 The visual test statistics $V_1(\mathbf{Y})$ and $V_2(\mathbf{Y})$ used. $V_1(\mathbf{Y})$ is a horizontal jittered dot plot while $V_2(\mathbf{Y})$ is a scatterplot of the first and second dimensional projections, with color representing groups in both cases.

placed randomly among the 19 null plots. To maintain the same orientation of the two groups in the 1D projection lineup, the mean of the projections for each group is calculated for each plot in the lineup and the group with the lower mean is considered to be group 1 and the other group 2. Figure 2.6 shows an example lineup having treatment levels $p = 20$, separation = Yes and $d = 1$. Similarly, Figure 2.7 shows an example lineup for $p = 100$, separation = No and $d = 2$.

A statistic measuring the ratio of the average distance within clusters to the average distance between clusters (Hennig, 2010), called WBratio, is calculated for each plot in the lineup of both 1D and 2D projections. An additional statistic Wilk's λ [e.g. (?)] is calculated for 2D projections. To account for the occasional lack of convergence of the projection pursuit optimization, 30 null plots are generated. The 19 null plots which have the smallest Wilk's λ values are used for the lineup.

2.6.4 Data collection

Subjects for the experiment were recruited through Amazon's Mechanical Turk (Amazon, 2010). Each subject was shown a block of ten lineups. They were asked to identify the plot which has the most separation between the colored groups. Their response was recorded along with a reason for their choice of the plot and the level of confidence they have in their decision. Gender, age, educational level and the geographic location of each subject were also noted. In total, 1137 lineups

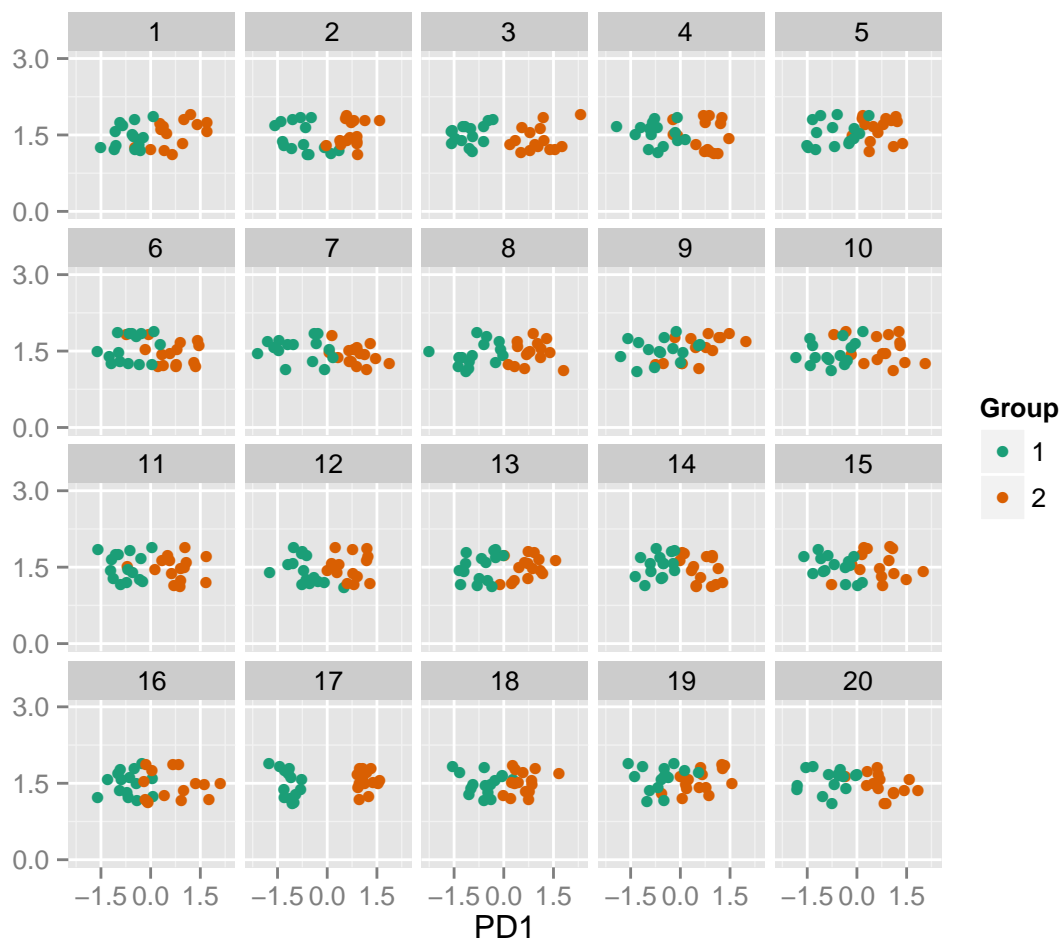


Figure 2.6 Lineup ($m = 20$) from treatment with $p = 20$, separation = Yes and $d = 1$. The subjects were asked to identify the plot with the most separated colors. Can you identify the observed data plot? The solution to the lineup is provided in the Appendix.

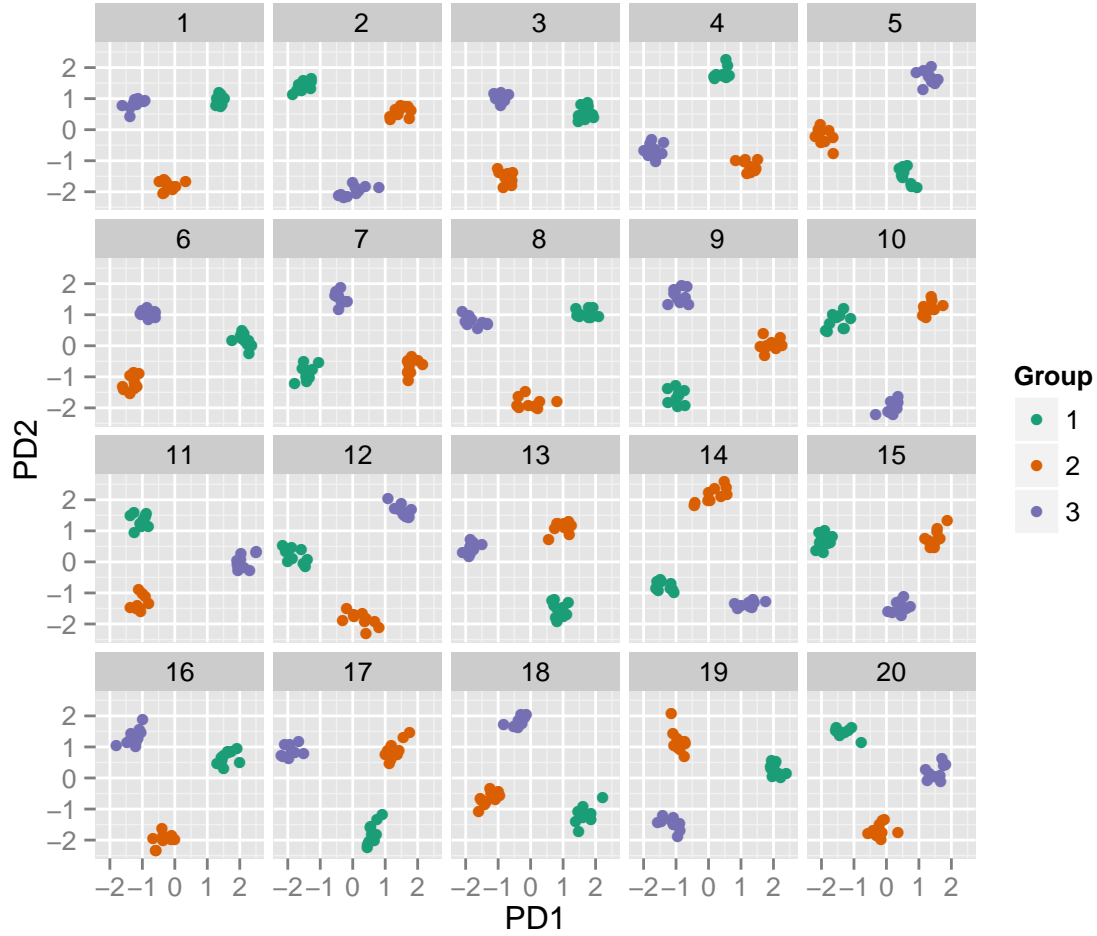


Figure 2.7 Lineup ($m = 20$) from treatment with $p = 100$, separation = No and $d = 2$. The subjects were asked to identify the plot with the most separation between the colored groups. Can you identify the observed data plot? The solution is provided in the Appendix.

were evaluated by 103 subjects, from different locations across the globe.

Each subject was given a very easy lineup (a lineup with $p = 10$ dimensions with some real separation) in the block of ten. Data from subjects who failed to give a correct response to this lineup are removed from the study. If their response to this lineup was correct, data for this lineup was removed but responses for the remaining nine were kept for analysis. This produced 66 lineups evaluated by 101 subjects for analysis.

2.7 Results

2.7.1 Effect of experimental factors on detection rate

We would expect that subjects detect the observed data plot more often when there is real separation but that the detection rate diminishes as dimension increases. This is indeed the case, as illustrated by Figure 2.8. The detection rate is plotted against data dimension (p), faceted by the levels of separation and projection. The three dots for each p represent the three replicates for each treatment. In a few cases, the dots overlap as the detection rate is same for the replicates. For example, when projection = 1D, dimension = 100 and separation = No, the detection rate is 0 for all the replicates and hence a single dot is shown. Alpha-blending is used on the plots so these ties are darker dots. The line shows the fixed effects from a logistic regression model fitted to the data. The detection rate is higher for small p , and on average decreases as p increases, with both 1D and 2D projections for real separation. For purely noise data, the detection rate is effectively flat across p , always less than 0.1 on average. Interestingly, the detection rate is higher for data where separation exists than for pure noise, even at $p = 100$, at 0.25 vs 0.05. For real separation, there also appears to be increasing variance as p increases, which is intriguing, too: it might be indicative of the increase in unexplained variance associated with the reduction to low dimensions from the increasingly higher p , and the sparsity of space.

Table 2.4 shows the estimates of the parameters from the fixed effects logistic regression model, the standard errors and the corresponding p -values. We observe that the p -values corresponding to dimension and presence of real separation is very highly significant. However, the p -value

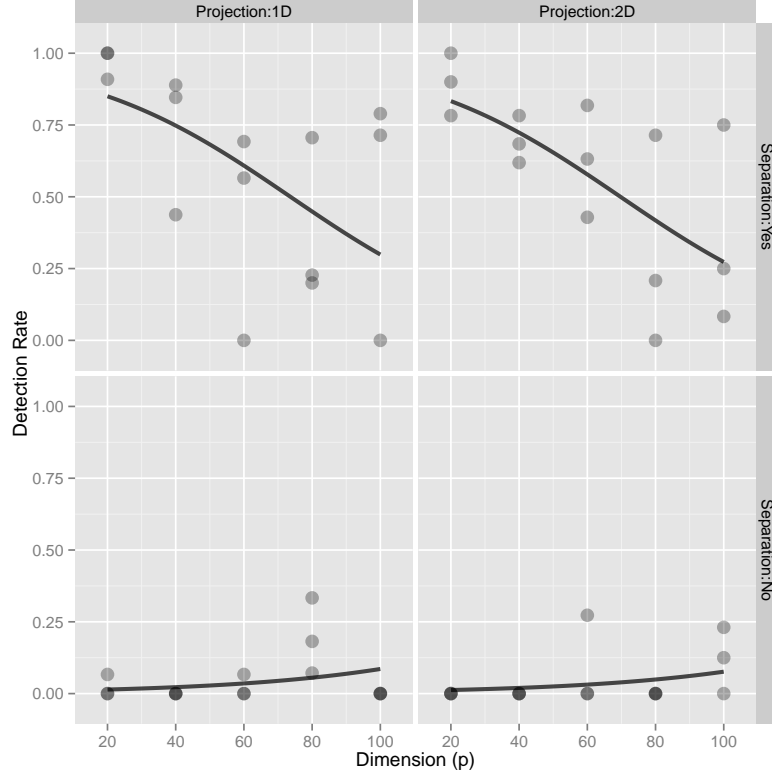


Figure 2.8 Detection rate by dimension, faceted by projection and separation. The three points represents the three replicates for each treatment level. A fixed effects logistic regression model is overlaid on the points. It can be seen that the detection rate decreases as p increases for data with real separation. When the data is purely noise data, the detection rate is flat across dimensions. Detection rate does not change with projection. Even with $p = 100$ subjects more often detected separation than would be expected by chance.

corresponding to the projection is large, which suggests that the difference between 1D and 2D projections is not significant. One of our concerns with the 2D projections is that the rotation of the group was not adjusted, and that this might diminish the subjects ability to identify the observed data plot. The lack of significant difference between 1D and 2D results suggest that rotation was not important. The interaction term is also significant, which says that the detection rate changes in the presence or absence of separation – detectability of the observed data plot decreases with dimension when there really is separation.

The effect of demographic variables (age, gender and education) on the detection rate was also

Table 2.4 Table summarizing results of experiment. Columns correspond to the estimate, the standard error and the p -value of the parameters used in logistic regression model. As dimension (p) increases, detection of separation decreases. Subjects can detect the separation if it exists even when $p = 100$. Subjects were equally good in 1D or 2D projections.

Parameters	Estimate	Std. Error	p -value
Intercept	2.381	0.278	0.000
dimension(p)	-0.032	0.004	0.000
separation = No	-7.097	0.911	0.000
projection = 2D	-0.127	0.181	0.483
separation:dimension	0.056	0.011	0.000

studied, and found to be insignificant.

2.7.2 Time taken to respond under different treatments

We would expect that the amount of time taken to respond will increase with the difficulty of identifying the observed data plot in a lineup. Figure 2.9 shows the time taken in seconds to respond (on a log scale) by p , faceted by projection. Color indicates presence or absence of separation. The line shows the trend over dimension. Bootstrap resampling bands (Buja and Wolfgang (2005)) are overlaid, which effectively provide confidence bands for the curves. Notice that, when the data has some real separation (green), as the dimension increases on average, subjects take more time to respond to the lineups. But when the data is purely noise (brown), the increase of dimension does not have any effect on the time taken. This suggests that as the number of dimensions increases, it becomes harder to spot the observed data plot among the null plots. On the other hand, the difficulty of spotting the observed data plot for a data with purely noise does not vary with dimension. It can also be seen that the time taken when the data is purely noise is overall higher than the time taken when the data has some real separation. The bootstrap resampling bands suggest that there is only significantly reduced time taken when $p = 20$ or 40.

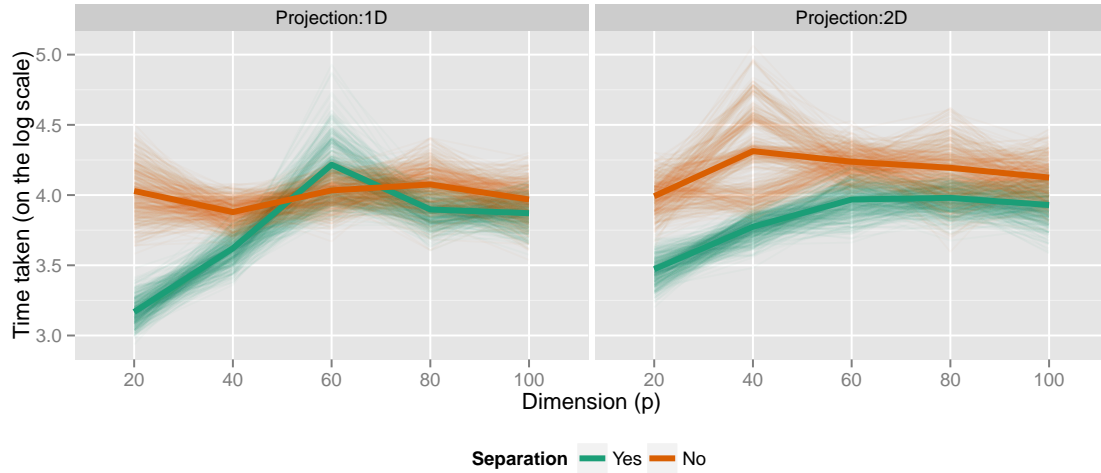


Figure 2.9 Time taken in seconds to respond on log scale against dimension colored by separation and faceted by projection. A line shows the trend over dimension for each separation within projection. Bootstrap resampling bands are drawn for each colored lines. Time taken to respond is higher when the data has no separation. Also as dimension increases, the time to answer when there is separation is equal to the time taken when there is no separation.

2.7.3 What affects decisions?

Figure 2.10 examines the subjects choices in detail. The relative frequency of picks of each plot in the lineup is plotted against a measure of average separation between groups. Each cell of this figure shows data from one of the lineups used in the study, 60 in total. Each “pin” represents a plot in a lineup, so each cell here has 20 pins, indicating the frequency that the plot was chosen. Red represents the observed data plot. Two separate figures are made for the 1D and 2D projections. The top three rows correspond to data containing real separation between the groups, and for the bottom three rows all of the data is purely noise. Columns indicate dimension (p). Replicates are in different rows. The taller the pin the more often that particular plot is chosen from the lineup. We asked subjects to pick the plot where the groups are most separated, and this is effectively what they picked. The plot in each lineup with the largest average separation tends to have the highest frequency. This is more obvious when there is real separation, and also when dimension is small, but it is also seen in the lineups containing pure noise data. This is reassuring – that subjects did

well at detecting the biggest difference.

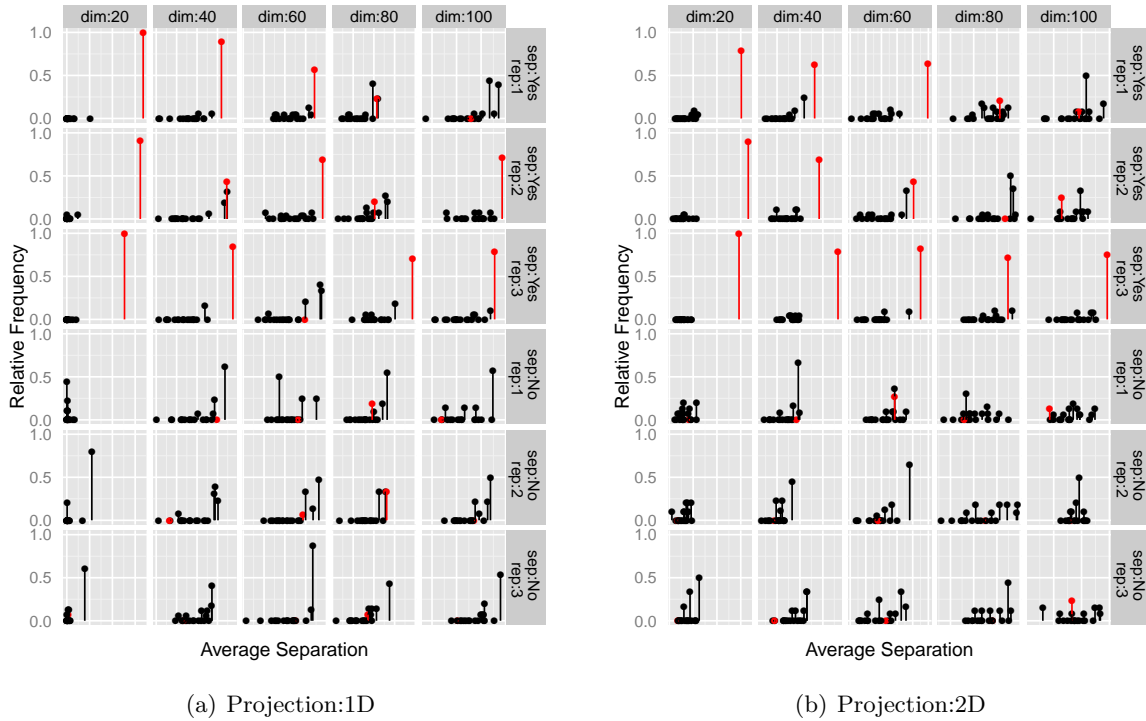


Figure 2.10 Comparing the choices that subjects make for each lineup. Relative frequency of plots chosen against a measure of the average separation between groups, the larger the value the more separated are the groups. Each cell here shows the data for one of the lineups used in the experiment, 60 in total, and each “pin” represents a plot in the lineup, 20 for each lineup. Red indicates the observed data plot. Subjects are asked to pick the plot in the lineup where the groups are the most separated, so we would expect that more subjects would pick the plots with the largest average separation. In general, this happens, the tallest pins are in the right of each cell. The top three rows show the results for the data with separation, so the observed data plot (red) is typically the pin on the very left of the cell, less so for the higher dimensions which are the cells at right. Figure (a) shows 1D projections and Figure (b) shows for 2D projections. There is not much difference between the two figures.

2.7.4 How do the null plots affect choices?

We have learned that subjects tend to pick the plot in the lineup that exhibits the most separation. Because visual inference only allows for a finite (small) number of comparisons against the sampling distribution, the influence of the null plots in the lineup on the observer’s choice is important. If

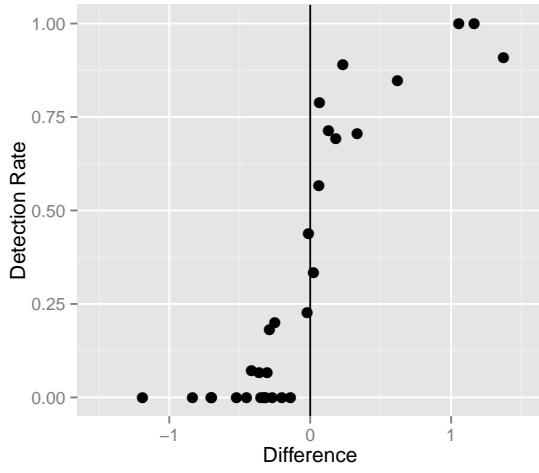
any null plot has a strong signal, subjects may choose this plot over the observed data plot. To gauge the influence of the null plots, we first calculate the average separation between the clusters in each plot of the lineup. For each lineup we then calculate the difference between the maximum average separation of the null plots and the average separation for the observed data plot for each lineup. Figure 2.11 examines the influence of the null plots on this pick. The detection rate and mean time taken in seconds are plotted against difference. The vertical line is a reference line where the difference is 0 – the value at which the observed data plot has the same signal as the most extreme null plot. The points to the right of the line should indicate easier lineups and those to the left indicate more difficult lineups in the sense that the null plots have more signal than the observed data plot. We can see that as the difference increases, the detection rate increases and also time taken to choose decreases, suggesting easier lineups. More details on the measurement of the influence of the null plots are available in Roy Chowdhury et al. (2012).

2.8 Conclusions

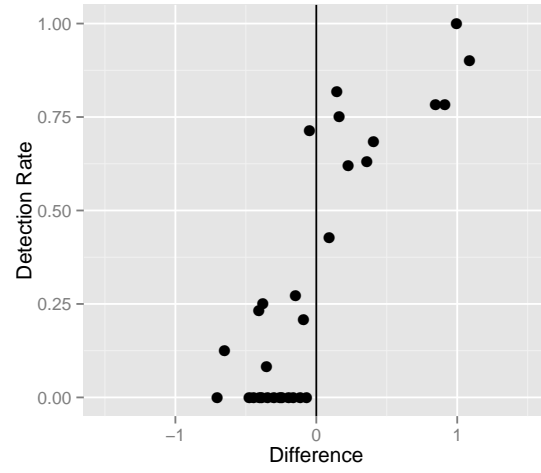
The results of applying visual inference procedures to classification problems on HDLSS data suggest that visual inference may be effective for improving the understanding of the emptiness of space in this type of data. With visual inference we saw that people can visually detect real separation as different from noise up to a reasonably high dimension, for 1D and 2D projections. Visual inference provides a calibration for reading the separation.

We also learned from visual inference, although didn't discuss this in the paper, that the projection pursuit optimization procedure in `tourr` package is performing correctly. It is possible that visual inference might be used to calibrate results of similar algorithms, where the optimization is used to yield visual products, like multidimensional scaling, PCA, independent component analysis (ICA) (Comon (1994)) and local linear embeddings (Roweis and Saul (2000)).

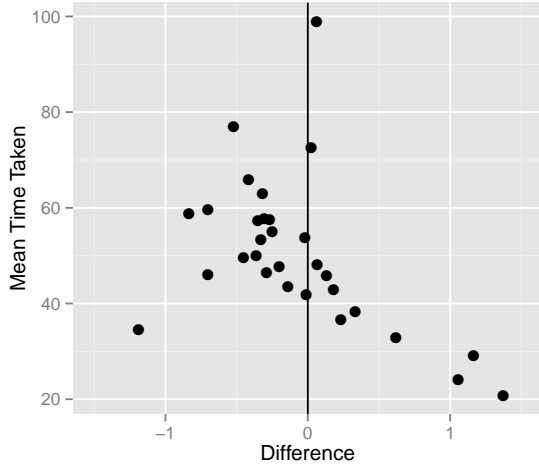
There are several natural next steps for this research. One is to examine the possibility of using visual inference to obtain confidence bands for the value of p , where separation is certain, for fixed sample size and dimension, particularly if a component of real separation is included. Another



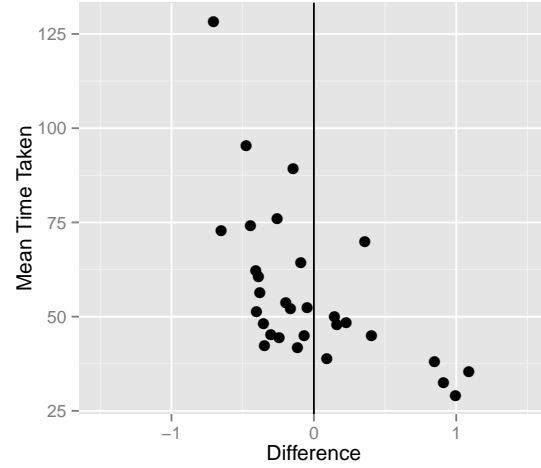
(a) Projection:1D



(b) Projection:2D



(c) Projection:1D



(d) Projection:2D

Figure 2.11 Detection rate and mean time taken to respond in seconds are plotted against the difference for 1D and 2D projections separately. The difference is between maximum separation of all the null plots and separation of the observed data plot for each lineup for 1D projections but for 2D projections the difference is based on the average separation between the groups. The vertical line represents difference equal to 1 when the average separation of the observed data plot is equal to the maximum average separation of the null plots for 2D projection. The points left to the line indicates a difficult lineup in the sense that at least one of the null plots had a lower average separation value than the observed data plot. (a) and (b) As difference increases, detection rate increases. (c) and (d) As difference increases, mean time taken decreases indicating that the subjects have an easier time in identifying the observed data plot.

direction is to build metrics to quantify the difficulty of a lineup and the influence that null plots have on identifying the data plot. It may be useful to incorporate the approaches into the statistics curriculum, particularly elementary statistics and applications areas such as gene expression analysis, to improve understanding of randomness.

Acknowledgement

This work was funded by National Science Foundation grant DMS 1007697. All figures were made using the R (R Core Team, 2013) package ggplot2 (Wickham, 2009).

CHAPTER 3. UTILIZING DISTANCE METRICS ON LINEUPS TO EXAMINE WHAT PEOPLE READ FROM DATA PLOTS

A paper to be submitted to *Journal of Computational and Graphical Statistics*.

Niladri Roy Chowdhury, Dianne Cook, Heike Hofmann, Mahbubul Majumder, Yifan Zhao

Abstract

Graphics play a crucial role in statistical analysis and data mining. This paper describes metrics developed to assist the use of lineups for making inferential statements. Lineups embed the plot of the data among a set of null plots, and engage a human observer to select the plot that is most different from the rest. If the data plot is selected it corresponds to the rejection of a null hypothesis. Metrics are calculated in association with lineups, to measure the quality of the lineup, and help to understand what people see in the data plots. The null plots represent a finite sample from a null distribution, and the selected sample potentially affects the ease or difficulty of a lineup. Distance metrics are designed to describe how close the true data plot is to the null plots, and how close the null plots are to each other. The distribution of the distance metrics is studied to learn how well this matches to what people detect in the plots, the effect of null generating mechanism and plot choices for particular tasks. The analysis was conducted on data that has already been collected from Amazon Turk studies conducted with lineups for studying an array of data analysis tasks.

3.1 Introduction

Graphics are an important component of big data analysis, providing a mechanism for discovering unexpected patterns in data. Pioneering research by Gelman (2004), Buja et al. (2009) and Majumder et al. (2013) provide methods to quantify the significance of discoveries made from visualizations. Buja et al. (2009) introduced two protocols which bridge the gulf between traditional statistical inference and exploratory data analysis. These are the Rorschach and the lineup protocols. The Rorschach protocol helps to understand the extent of randomness. The lineup protocol places a statistical plot firmly in the hypothesis testing framework, where a plot of the data is considered to be a test statistic. Unlike the simpler numeric test statistics in classical inference, though, the plot as a test statistic is a complex entity. This plot is compared with a set of null plots, obtained from an appropriate distribution consistent with the null hypothesis. The lineup protocol places the data plot randomly among the obtained null plots, and requires a human observer to examine the plots and identify the most different plot. If this plot is that of the data, this is quantifiable evidence against the null hypothesis.

Figure 3.1 is an example of a lineup. Suppose we have the following statistical model

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \epsilon_i$$

and we are interested in testing the following hypothesis:

$$H_o : \beta_k = 0 \quad \text{vs} \quad H_A : \beta_k \neq 0$$

where X_k is a continuous covariate. The true plot is obtained by plotting Y against X_k with a regression line overlaid. The null plots are obtained by simulating data from $N(X\hat{\beta}, \hat{\sigma}^2)$ and plotting using the same scatterplot method as the true data. These parameter estimates $(\hat{\beta}, \hat{\sigma}^2)$ are obtained by fitting the null model to the true data. The plot of the true data is randomly placed among a set of $(m - 1)$ null plots to produce a lineup of size m . The human subjects are then shown this lineup and asked to identify the plot which has the steepest slope. If the human subjects can identify the

plot of the true data, we reject the null hypothesis and conclude that there is a significant linear relationship between Y and X_k .

The lineup protocol was formally tested in a head-to-head comparison with the equivalent conventional test by Majumder et al. (2013). The experiment utilized human subjects from Amazon’s Mechanical Turk (Amazon (2010)) and used simulation to control conditions. The results suggest that the visual inference is comparable to conventional tests in a controlled conventional setting. This provides support for its appropriateness for testing in real exploratory situations where no conventional test exists. Interestingly, the power of a visual test increases with the number of observers engaged to evaluate lineups, and the pattern in results suggests that the power will provide results consistent with practical significance (Kirk (1996)).

In traditional hypothesis testing, the sampling distribution of a test statistic is functional and continuous. In the lineup protocol, although conceptually we may have an infinite collection of plots from the null distribution, in practice, we can only evaluate against a finite number of null plots. A human judge has a physical limit on the number of plots they can peruse. This poses one of the issues with using the lineup protocol. Figure 3.2 illustrates the difference. In traditional inference, the black curve represents the sampling distribution for the t -distribution under the null hypothesis, and the shaded red area shows the rejection region. In visual inference, let us consider that the black curve gives the sampling distribution although the sampling distribution is essentially a distribution of null plots. Although the test statistic is not numeric, the true data plot which is the test statistic is represented using red bar and the null plots that are drawn from the null distribution are the blue bars. Effectively, in visual inference the red line is compared only to these finite number of blue lines visually to make a decision, unlike classical inference where we look at the rejection region (Figure 3.2) to make decisions. Even though the data plot might be extreme, it is possible by randomly selecting from the null distribution, to obtain a null plot that is more extreme, as Tukey suggested (Fernholz, 2003):

“There [in Tukey’s Data Analysis class] I discovered that [...] a random sample is indeed a “batch of values” which “fail to be utopian” most of the time.”

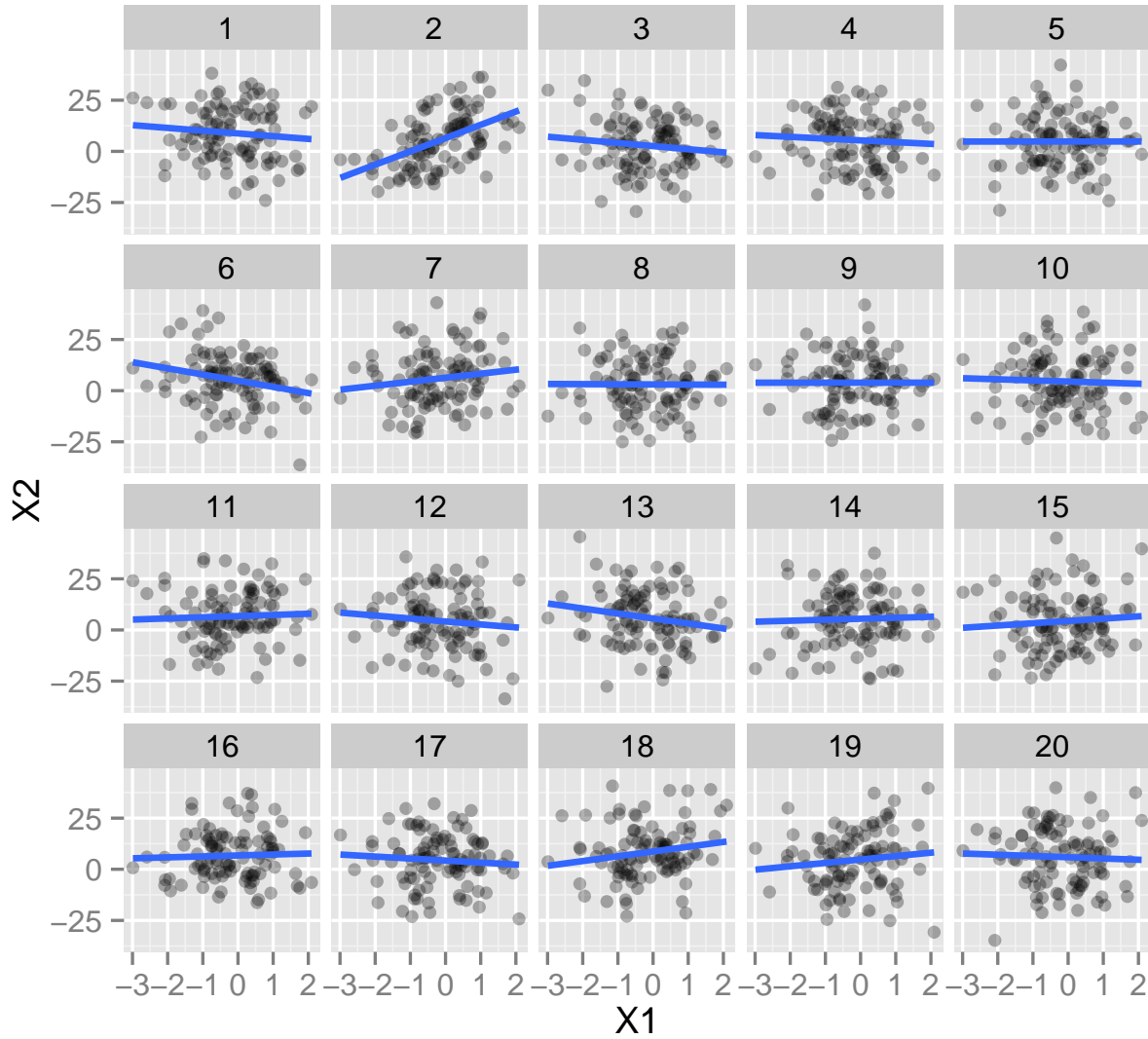


Figure 3.1 Lineup plot of size $m = 20$ using scatterplots with regression line overlaid. This tests $H_o : \beta_k = 0$, where covariate X_k is continuous. One of the plots in the lineup is the plot of the true data. The other plots are null plots generated by simulating data from a null model that assumes that the null hypothesis is true. Can you identify the plot with the steepest slope?

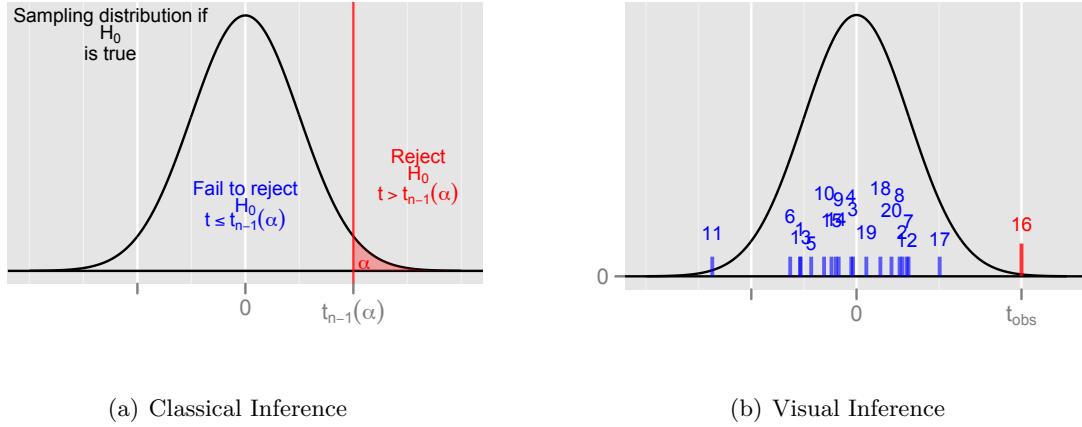


Figure 3.2 If the lineup protocol was to be used instead of classical inference this is what it would look like. (a) Decision region (shaded in red) for classical inference for $H_0 : \mu = \mu_0$ vs $H_a : \mu > \mu_0$ and (b) values corresponding to the true value (red) and the null plots (blue) in a single lineup of size $m = 20$ that would be used to test the same null hypothesis. The actual data plot is extreme relative to the null plots, and observers would likely be able to pick it out, resulting in a decision to reject the null hypothesis. In practice, the lineup protocol would not be used if a classical test can be used.

This can be partially solved by having a large number of observers, who each evaluate lineups constructed with different null plots. Having some idea of the type of coverage of the sampling distribution that is provided by the lineups would be useful ahead of engaging observers and evaluating the lineups. Could we say that lineup X is expected to be “difficult” but lineup Y is expected to be “easy” then it may help in determining an appropriate number of observers? A difficult lineup is one where the data plot is similar to the null plots, and an easy lineup is where the data plot has some feature that makes it very different from the null plots. Being able to compute a plot to plot distance metric would be very helpful ahead of running a lineup protocol.

This is a two way process: As metrics are devised to measure the quality of a lineup, the lineup protocol also provides an opportunity to measure the performance of a metric. The human eye can detect patterns in a plot that just cannot be easily quantified numerically, which is why graphics provide an important tool for exploring data and finding the unexpected. Describing plots numerically, is something of an oxymoron, it cannot be universally done. An example in past work are scagnostics (Tukey, 1977; Wilkinson et al., 2005) which were developed to assess the different

aspects of scattered points like outliers, shape, trend, density and coherence. If a scatterplot has just one of these structures the scagnostics are descriptive, however, they fail terribly if a plot contains more than one. The goal here is to find some distance measures that can provide some indications of the quality of a lineup, and then to use the results of observer evaluation to determine which metrics best match what people see.

Following up on choices, observers are asked to describe their reasoning. These reasons are used to obtain more information about the rejection: was it some nonlinear dependency, an outlier, clustering, that triggered the detection of the data plot? Good distance metrics may also help relate the descriptive words used with mathematically defined features.

The article is organized as follows. Section 3.3 discusses the null generating mechanisms. Section 3.4 defines the distance measures and discusses the choice of the measures. The distribution of the distance measures are studied in Section 3.5. Section 3.6 describes the effect of the plot type and the question of interest on the distance measure while Section 3.7 talks about the distance evaluations. In Section ??, the methods to select the number of bins for the binned distance is described. Section 3.8 presents a comparison of the distance measures to the performance of human subjects in several experiments conducted by Amazon’s Mechanical Turk.

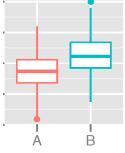
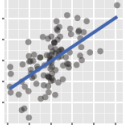
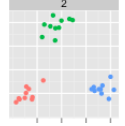
3.2 Experimental Data

There have been eleven experiments conducted using Amazon’s Mechanical Turk (Amazon, 2010) (MTurk), with some used for evaluating the lineup protocol against classical testing, and the others using the protocol for different purposes. It is possible to try the tasks provided to Turkers (?). For evaluating the distance metrics we used the data collected on experiments 1, 2, and 7. Table 3.1 describes the experiments.

3.3 Null Generating Mechanism

The lineup protocol embeds the true data plot among a set of null plots. The method of obtaining the data for these null plots is called the null generating mechanism. The null hypothesis directly

Table 3.1 Overview of the different Turk experiments, from where data was taken to study distance metrics and how subjects read the plots.

Turk ID	Experiment	Test Statistic	Lineup question
1	Box plot		Which set of box plots shows biggest vertical difference between group A and B?
2	Scatter plot		Of the scatter plots below which one shows data that has steepest slope?
7	Group separation		Which of these plots has the most separation between the coloured groups?

affects the choice of null generating method. In the experimental data that we analyzed the null generating methods used were:

- **Permutation:** This is the most commonly used approach thus far, because it can be used in a variety of problems. Permutation is used to break association between two or more variables, and thus is appropriate when the null hypothesis is that there is no association. Consider two variables X_1 and X_2 . Either X_1 or X_2 is permuted keeping the other variable fixed. Any association between X_1 and X_2 is broken in the process. The marginal distribution of X_1 and X_2 remains the same while the joint distribution is altered. The method works in situations where one or both the variables are continuous or categorical. Let us consider a case where we have one categorical variable, say, Group and a continuous variable. Let us assume that the variable Group has two levels (say, A and B) and we want to test whether there is any significant difference between the two groups, i.e. $H_o : \mu_A = \mu_B$. To generate the null data, the values of the variable Group are permuted keeping the continuous variable fixed. If there is a difference between the two groups, this difference is broken by the permutation, and any difference observed in the permuted data is consistent with random variation.
- **Simulation under a null model:** Sometimes there is a model underlying the problem being

studied. In this situation simulating from the model will be the null generating mechanism. Assuming that the null hypothesis is true, the model is fitted to the true data. The parameter estimates are obtained from the fitted model and then the data is generated using the parameter estimates. Let us consider that we are interested in testing whether there is any significant linear relationship between two continuous variables X_1 and X_2 . Hence we test for $H_o : \beta_1 = 0$ versus $H_a : \beta_1 \neq 0$. Under the null hypothesis, we fit the following model to the data:

$$Y = \beta_0 + \varepsilon$$

where $\varepsilon \sim \text{Normal}(0, \sigma^2)$. The parameter estimates of β_0 and σ^2 are obtained and the null data is generated from $\text{Normal}(\widehat{\beta}_0, \widehat{\sigma}^2)$.

3.4 Distance Measures

By calculating the “distance” between plots we may be able to determine if a lineup should be easy – the the actual data plot is detectably different from the null plots – and also to better understand what aspect of the plot people use to make their choice. It is not an easy task to measure the difference between plots. Here we examine several possibilities.

The problem could be tackled by considering the data as a reference distribution, and compare all of the null sets with this reference. Comparing data with a reference probability distribution or comparing two datasets are common statistical tasks. For example, the Kolmogorov-Smirnov test (?) sorts values in two samples, computes the empirical distribution function of each and compares these two, to determine if the two samples are likely to have come from the same distribution. The Anderson-Darling (?) and Shapiro-Wilk (?) tests compare datasets with normal probability distributions. These measure differences between univariate distributions which limits their applicability to distances between plots, generally.

Hausdorff distance (Huttenlocher et al., 1993) has been successfully used for comparing images. It effectively matches points between sets and computes the distances between the matched points. When permutation is the null generating mechanism, Hamming distance (?) can be used to calculate

how different the permutations are, by measuring the minimum number of substitutions it takes to get from one permutation to another.

Alternatively, interpoint distance metrics might be adapted to measure distances between plots. For example, when the purpose is differences between groups in a single plot, like side-by-side boxplots, a distance metric that focuses on group separation calculated on each data set might be useful. Bhattacharyya distance (?) is widely used in image processing, for feature extraction. The R package `fpc` Hennig (2010) contains many different ways to calculate distances between groups.

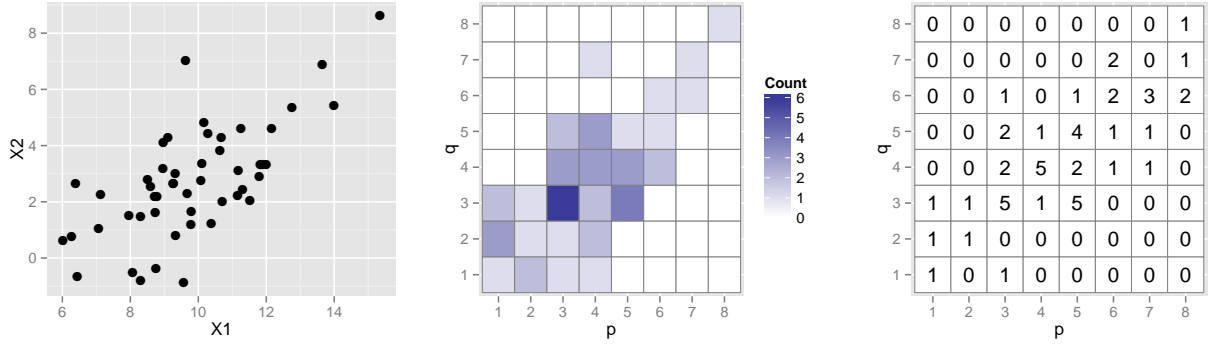
Ultimately, a very simple binned distance was used in the analyses of the MTurk data, which worked fairly well in most circumstances. However, it was clear immediately that plot design, and the question asked, has a large impact on how a plot is read, and specific distance metrics designed for specific plot types is needed. Below is a summary of distance metrics used. For all of the distance measures below, let X denote the true dataset with one or two variables. Let Y denote the null dataset obtained from X using an appropriate null generating mechanism.

- Binned Distance (BN): Let X_1 and X_2 be two continuous variables. Let X_1 be divided into p bins and X_2 divided into q bins. Let $C(X_1, X_2)$ be defined as a $p \times q$ matrix. Each (i, j) -th element of the matrix represents the number of points falling in the (i, j) -th cell, where $i = 1, \dots, p$, $j = 1, \dots, q$. The distance is then defined as

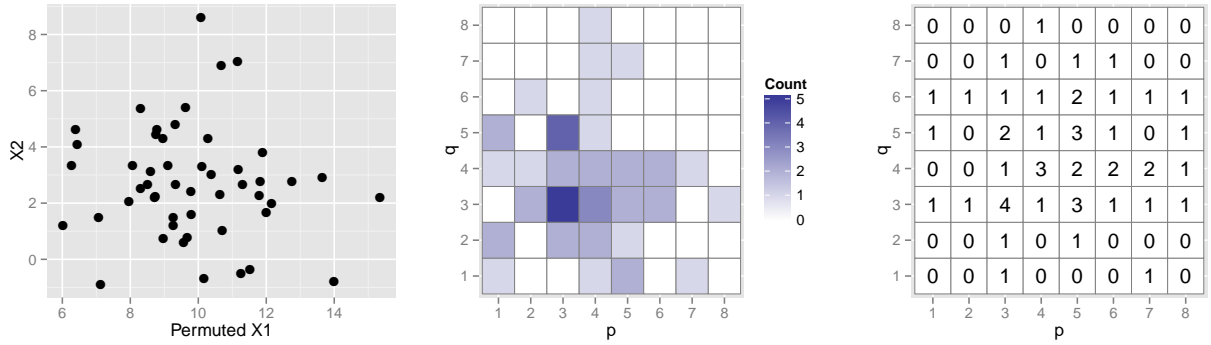
$$\begin{aligned} d_{\text{BN}}^2(X, Y) &:= \|C_X(X_1, X_2) - C_Y(X_1, X_2)\|^2 \\ &= \sum_{i=1}^p \sum_{j=1}^q (C_X(X_{1i}, X_{2j}) - C_Y(X_{1i}, X_{2j}))^2. \end{aligned}$$

This distance can be calculated for univariate continuous data, bivariate data with two categorical variables, or data with one continuous and one categorical variable. For the categorical variable, the number of bins would equal to the number of categories.

Binned distance is highly susceptible to small differences in values and depends on the number of bins as well the anchor positions. It is necessary to find the optimal number of bins in each direction. For our purposes optimal was defined number of bins that produced the most detectable observed data plot. The distance was measured between the observed data plot,



(a) Dataset X with two variables X_1 and X_2



(b) Dataset Y with permuted X_1 and original X_2

Figure 3.3 Illustration of binned distance, for data with strong association (a), and the same data where one variable has been permuted (b). The scatterplot of the data is shown (left) along with the binned view of the data (center) and the number of points in each cell (right). Binned distance is the euclidean distance of these counts. The binned distance between these plots is 6.4807.

and the closest null, and compared with the biggest distance between any pair of null plots. Details of these choices on various different data sets is in the Appendix.

Variations on this distance are possible, using kernel density estimates, or using a different power than the square, or using transformations on the counts. Hausdorff distance (Huttenlocher et al., 1993) was examined as a generic distance metric as well, but the binned distance was faster to calculate and performed as well as the Hausdorff as a rough, generic measure of similarity of plots.

- Distance based on boxplots (BX): Let X_1 be a categorical variable representing the groups in the data and X_2 be a continuous variable. Then the distance metric is given by

$$d_{\text{BX}}^2(X, Y) := \|d_q(X) - d_q(Y)\|^2 = \sum_{i=1}^3 ((d_q(X))_i - (d_q(Y))_i)^2$$

where $d_q(\cdot)$ is a vector giving the absolute difference of the first quartile, median and the third quartile of X_2 between the two groups in X_1 . This distance measure works specifically for the boxplots using only the graphical elements. This is based on the assumption that after the boxplots have already been constructed, the subjects only look at the difference in the boxes to make the distinction. Variations on this might include adding whiskers ends, outliers, or even removing the absolute value.

- Distance based on the regression line (RG): Let X_1 and X_2 be two continuous variables. X_1 and X_2 are plotted in a scatterplot and assume that the scatterplot is binned vertically into b bins. In each vertical bin, a linear regression model is fitted and the regression coefficients i.e. the estimated intercept and the estimated slope are noted. The distance metric based on the regression coefficients is given by

$$d_{\text{RG}}^2(X, Y) := \text{tr}(B(X) - B(Y))'(B(X) - B(Y)) = \sum_{i=1}^b ((b_0(X))_i - (b_0(Y))_i)^2 + \sum_{i=1}^b ((b_1(X))_i - (b_1(Y))_i)^2$$

where b_0 and b_1 denote the vector of the intercept and slope respectively while b is the number of bins. $B(\cdot)$ is a $b \times 2$ matrix of the regression coefficients where each row represent the intercept and the slope obtained from each bin. The number of bins have a significant effect

on the distance measure. It can be seen that it works best for smaller number of bins like 1 or 2. With larger number of bins (i.e. smaller bin sizes), the regression coefficients are affected by the skewness of the data. Variations might include using slope alone, or absolute value of slope.

- Distance based on separation between multiple groups (MS, AS, CM): Let X_1 and X_2 be two continuous variable. Let X_3 be a categorical variable providing the groups associated with each variable. X_1 and X_2 are plotted in a scatterplot colored by the group variable X_3 . The separation can be described in a number of different ways (Hennig (2010)). Two versions are used in this paper. Let us define,

(i) $s_m(\cdot)$ be a vector of cluster wise minimum distance between a point in the cluster to the points in other clusters for g clusters. The distance metric based on separation is defined as

$$d_{\text{MS}}^2(X, Y) := \|s_m(X) - s_m(Y)\|^2 = \sum_{i=1}^g ((s_m(X))_i - (s_m(Y))_i)^2$$

(ii) $s_a(\cdot)$ be a vector of cluster wise average distances of all the points in the cluster to all point of other clusters for g clusters. The distance metric based on separation is defined as

$$d_{\text{AS}}^2(X, Y) := \|s_a(X) - s_a(Y)\|^2 = \sum_{i=1}^g ((s_a(X))_i - (s_a(Y))_i)^2$$

(iii) Let $m_i(\cdot) = (\bar{X}_1^i, \bar{X}_2^i)$ be the cluster mean for the i -th cluster, $i = 1, 2, \dots, g$. The distance metric based on the cluster means is given by

$$d_{\text{CM}}^2(X, Y) := \|m(X) - m(Y)\|^2 = \sum_{i=1}^g (m_i(X) - m_i(Y))^2$$

Figure 3.4 illustrates the difference between the different distance metrics separation. In practice, many possible metrics could be used to measure the separation, such as those readily available in the **fpc** package.

How well each distance measure matches the observers' responses depends some on the question of interest. But, in general, this should not be a problem because the question which is typically asked is "Which plot among these is different ?". In the MTurk experiments, more focused questions were asked because the purpose was very specific, to compare visual inference with classical inference.

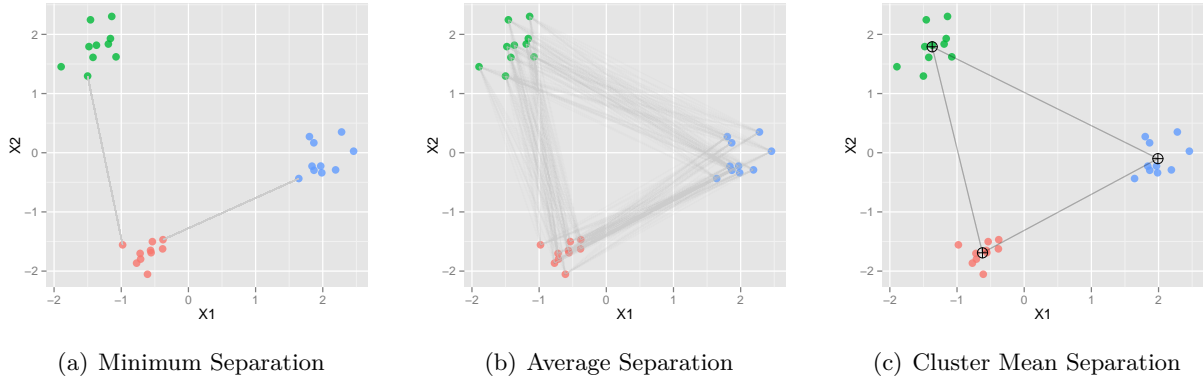


Figure 3.4 Illustration of three different distance metrics based on separation. Two dimensional projections are plotted with 3 groups. Minimum Separation (in (a)) calculates the minimum distance between points of each cluster from the other clusters. Average separation (in (b)) calculates the average distance of each point in a cluster to the other clusters. In (c), the cluster mean distance calculates the distance between the means of each cluster.

3.5 Distance Metric Distribution

For a given lineup of size m , the empirical distribution of distance metrics is obtained using the following algorithm:

1. Calculate the distance between the true data and all the null datasets and take the average of these distances.
2. For each of the $(m - 1)$ null datasets, calculate the distance between the null data and all the other $(m - 2)$ null datasets and obtain the average distance. Hence obtain $(m - 1)$ distances corresponding to each null plot.
3. Generate a null dataset from the true dataset by the null generating mechanism. Consider this null dataset as “true” dataset. From this “true” dataset, generate $(m - 2)$ null datasets and calculate the average distance between the “true” data and the null datasets. Hence obtain a distance corresponding to this “true” dataset. Repeat this procedure a large number of times (order of 1000 or 10000).
4. Plot these 1000 distances to obtain the empirical distribution of the distance metric. Plot the distances for the true plot and the null plots to show the distances for the lineups.

The empirical distribution of the distance measures is obtained by calculating the distances between the null plots among themselves. One null data is generated from the true data set using the null generating mechanism. Assuming this null data to be the “true” data set, a number of null data sets are obtained from this null data and the distances between these datasets are calculated. One single distance value is obtained by averaging all these distances. This process is repeated a large number of times, say, N where N is a large number of the order 10^3 or 10^4 . Finally N mean distances or average distances are obtained which gives the empirical distribution of the distance.

The empirical distribution of the distance works as the t -distribution in the classical setting. In the classical setting, the test statistics follows a t -distribution under the null hypothesis. The observed test statistic is then compared to this distribution, as shown in Figure 3.2. In visual inference, the mean distances of the null plots gives the empirical distribution. The mean distance of the true plot from the null plots in the lineup acts as the observed test statistic. Unlike the t -distribution, the empirical distribution is generally skewed.

The mean distance between the true plot and the null plots in a lineup of size $m = 20$ is calculated by averaging over the distances between the true plot and each of the $(m - 1)$ null plots. The mean distances for the $(m - 1)$ null plots in the lineup are calculated by taking the mean of the distances of the particular null plot and the other $(m - 2)$ null plots. The mean distances for the true dataset and the null datasets are plotted on the empirical distribution. If the mean distance of the true plot is larger than any of the null plots, the lineup would be regarded as “easy”. Otherwise, it is a “difficult” lineup.

The empirical distribution of the distance based on regression is shown in Figure 3.5. To generate this distribution, $N = 1000$ and $m = 20$ was used. Figure 3.5(a) shows the lineup plot for $m = 20$ for testing whether there exists a significant linear relationship between X_1 and X_2 . The 19 null plots are generated by fitting the null model and generating from the null model. Figure 3.5(b) shows the empirical distribution of the distance with the mean distances for the true plot (in orange) and the null plots (in black) for the particular. The true plot is easy to be identified in the lineup (Figure 3.5(a)). It can also be seen in Figure 3.5(b) as the orange line is extreme compared to the black lines.

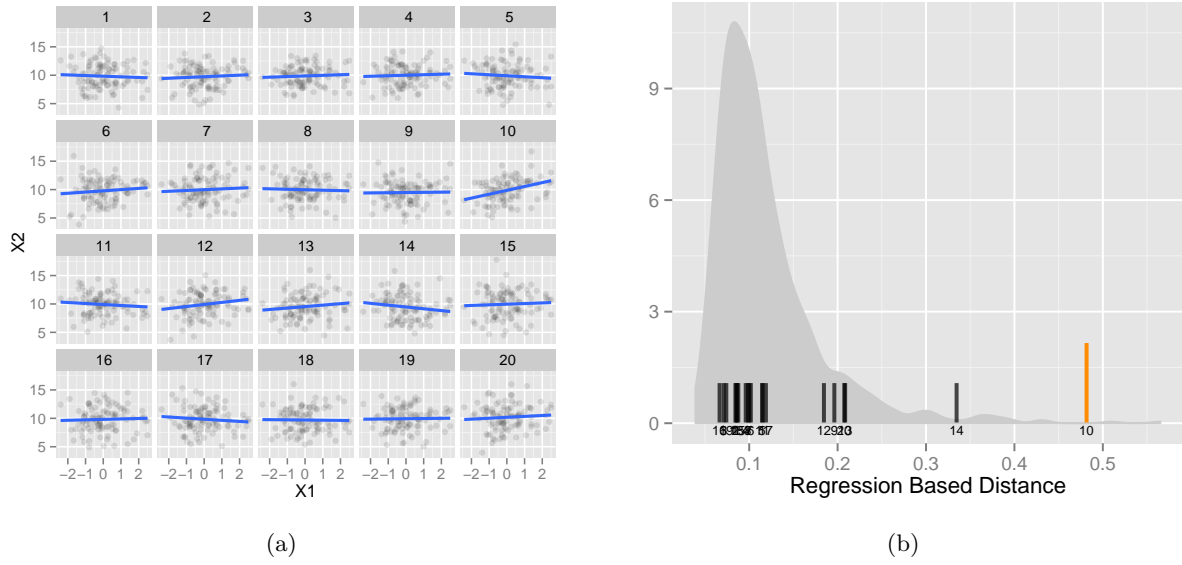


Figure 3.5 Illustration of the behavior of a distance metric with a lineup plot in (a) and the distribution of regression based distance metric in (b). A lineup of size $m = 20$ is shown (left) for testing whether there exists a significant linear relationship between X_1 and X_2 . The 19 null plots are obtained by simulating from the null model. The empirical distribution of the distance metric is shown (right). The distances for the true plot and the null plots are shown in orange and black respectively.

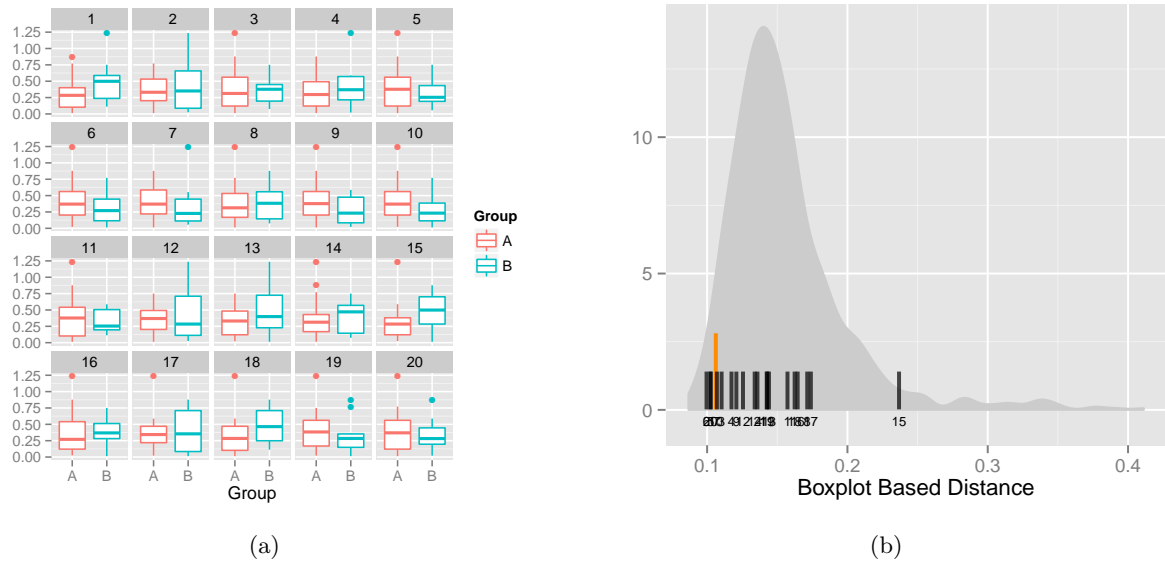


Figure 3.6 Illustration of the behavior of a distance metric with a lineup plot in (a) and the distribution of boxplot based distance metric in (b). A lineup of size $m = 20$ is shown (left) for testing whether there exists a significant linear relationship between X_1 and X_2 . The 19 null plots are obtained by simulating from the null model. The empirical distribution of the distance metric is shown (right). The distances for the true plot and the null plots are shown in orange and black respectively.

Figure 3.6(a) shows the lineup plot for $m = 20$ for testing whether there exists a significant difference between the two groups A and B. The 19 null plots are generated by permuting the group variable keeping the other variable fixed. Figure 3.6(b) shows the empirical distribution of the distance based on the boxplots with the mean distance for the true plot (in orange) and the null plots (in black). The true plot is hard to be identified from the lineup which is also evident in the distribution since many black lines are to the right of the orange line.

3.6 Effect of Plot Type and Question of Interest

Previous studies have suggested that the type of plot used in the lineup have an effect on the response of the subjects (Zhao et al., 2012). For example the subjects find it easier to identify the true plot for a large sample data when a box plot is used in the lineup instead of a dot plot. Similarly the distance metric should also be altered according to the plot type. The distance metric should account for the additional information provided by the graphical elements in the lineup. The graphical elements, like the presence of a box or a regression line overlaid on a scatterplot may influence the response of the subject. Figure 3.7 illustrates this idea.

Figure 3.7(a) shows a lineup of scatterplots with 100 points between two variables X_1 and X_2 . Figure 3.7(b), on the other hand, gives a lineup of the same scatterplots with the regression line overlaid. Showing Figure 3.7(a), if the subjects are asked to identify the plot which has the steepest slope, then the subjects probably will face some difficulty in identifying the true plot. But in Figure 3.7(b), the regression line overlaid makes it easier for the subjects to identify the true plot. A different distance metric should be used in each case to correctly measure the quality of the lineup.

The question asked to the subjects plays an important role to identify the true plot in the lineup. A minor change in the question can change the response of the subject. In Figure 3.6(a), if the subjects are asked to identify the plot in which the green group has a larger vertical difference than the red group, the subjects should pick Plot 6. If the subjects are asked which plot has the largest vertical difference between the two groups, the subjects should pick Plot 15. A distance metric should also take into account the question of interest. But, in general, the question of interest is

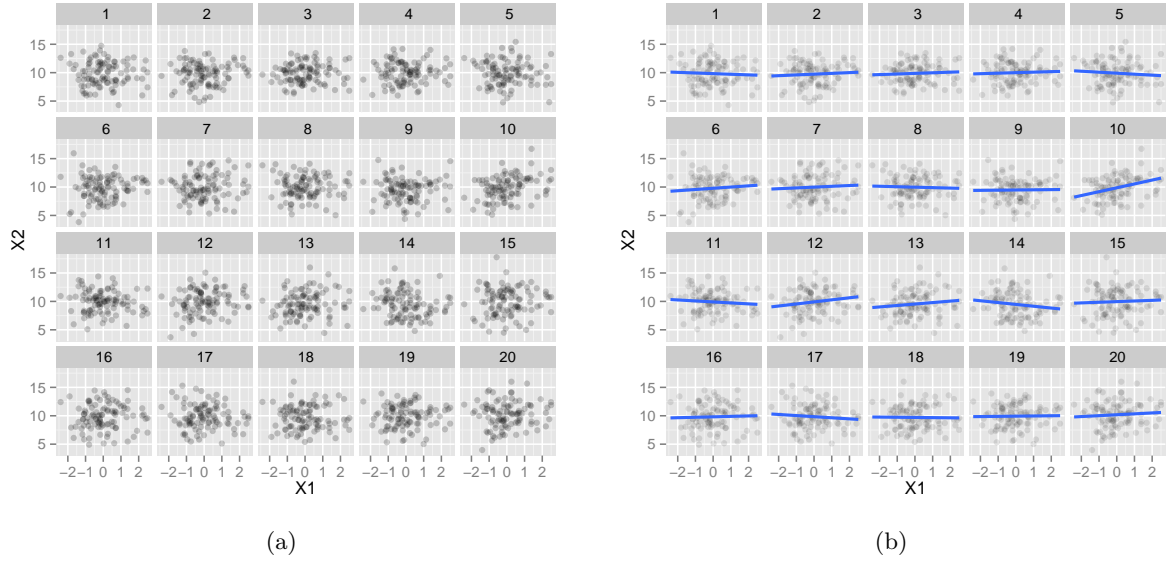


Figure 3.7 Comparison of two lineups: scatterplots in (a) and scatterplots with a regression line overlaid in (b). The raw data is same for the lineups. The human subjects are shown the lineups and asked to identify the plot with the steepest slope. The presence of regression line may affect the decision of human subjects.

which plot among these is different.

3.7 Metric Evaluation

For a lineup of size $m = 20$, the distance for the true plot is compared to the 19 null plots. This comparison can sometimes complicate things. A logical solution can be to look at one statistic for one lineup. Such a statistic can be defined as the difference between the mean distance of the true plot and maximum of the mean distances for the null plots. Hence we define,

1. Difference: the difference between the mean distance for the true plot and the maximum of the mean distances for the null plots. Mathematically,

$$\delta_{\text{lineup}} = \bar{d}_{\text{true}} - \max_j \bar{d}_{\text{null}_j}$$

for $j = 1, \dots, (m - 1)$. A positive difference would indicate that the mean distance of the true plot is greater than the maximum of the mean distances of the null plots. Hence the true plot

is extreme compared to all the null plots. Similarly a negative difference indicates that there is at least one null plot which is extreme compared to the true plot based on the distance.

The issue with this statistic is that δ_{lineup} indicates an “easy” or “difficult” lineup only on the basis of whether it is positive or negative, although it may be really close to 0. The statistic does not imply how many null plots are more extreme than the true plot. So we define,

2. Larger than the true plot: the number of null plots which have larger mean distances than the mean distance of the true plot is noted. Mathematically,

$$\gamma_{\text{lineup}} = \sum_{j=1}^{m-1} a_j$$

where

$$a_j = \begin{cases} 1 & \text{if } \bar{d}_{\text{null}_j} > \bar{d}_{\text{true}}, \\ 0 & \text{otherwise} \end{cases} \quad (3.1)$$

γ_{lineup} takes all values between 0 and $(m - 1)$. A large value of this measure would indicate that there are a number of null plots more extreme than the true plot and hence it is hard to identify the true plot in the lineup.

3.8 Results

The performance of the distance metrics was evaluated with comparing the distances with the response of the subjects. A number of experiments were done in Amazon Mechanical Turk (Amazon, 2010). Subjects were recruited through Amazon Mechanical Turk (Amazon, 2010) and were shown a sequence of lineups. In each experiment, they were asked specific questions. Their responses were recorded along with other demographic informations. The details about the design of experiments can be found in ?) and Roy Chowdhury et al. (2014).

3.8.1 Turk Experiment 1 – Side by Side Boxplots

In this experiment, all the lineups generated had a side by side boxplot as the test statistic. Assuming that the null hypothesis is true, the null plots were generated by assuming that there is

no difference between the two distributions. The subjects were shown a few lineups and were asked to identify the plot which has the largest vertical difference between group 1 and group 2. Figure 3.8 gives such a lineup.

The response of the subjects were noted and the proportion of correct response was calculated for each lineup. The distances between the plots in each lineup were computed using both the distance based on boxplots (d_{box}) and the binned distance (d_{bin}). The mean distance for the true plot and the null plots were calculated and δ_{lineup} and γ_{lineup} are obtained. The proportion of correct response was plotted against each of the two statistics. Figure 3.9 shows the detection rate against the difference for d_{box} and d_{bin} and the number of null plots greater than the observed plot for the two distance measures.

In Figure 3.9, the detection rate is plotted against the difference. The red vertical line represents difference equal to 0 indicating that the mean distance of the true plot is equal to the maximum of the mean distance of the null plots i.e. the mean distance of the true plot is equal to at least one of the mean distance of the null plots. It can be seen that as the difference increases, the detection rate increases. So the subjects do better in the easier lineups than the hard ones. The binned distance was calculated using 8 bins on both the axes. Figure 3.9 also shows the relation between detection rate and the number of null plots larger than the true plot. It can be seen that as there are more extreme null plots compared to the observed plot, the subjects find it difficult to pick the observed plot. It is interesting to see that the subjects can pick the observed plot with one or two extreme null plots.

Though the distance based on the boxplots works better, the binned distance does a decent job in this case. According to the binned distance, there are a few lineups which has a negative difference but the proportion correct is above 60%, which can be also be seen in Figure 3.9. It should be noted that the binned distance does not take into account the graphical elements of the plot (e.g. boxplot) and calculates the distance solely based on the data. So an outlier may have a huge effect on the binned distance but does not effect the distance based on the boxplots. Hence it is advisable to use a distance based on the graphical elements since that is exactly what the subjects

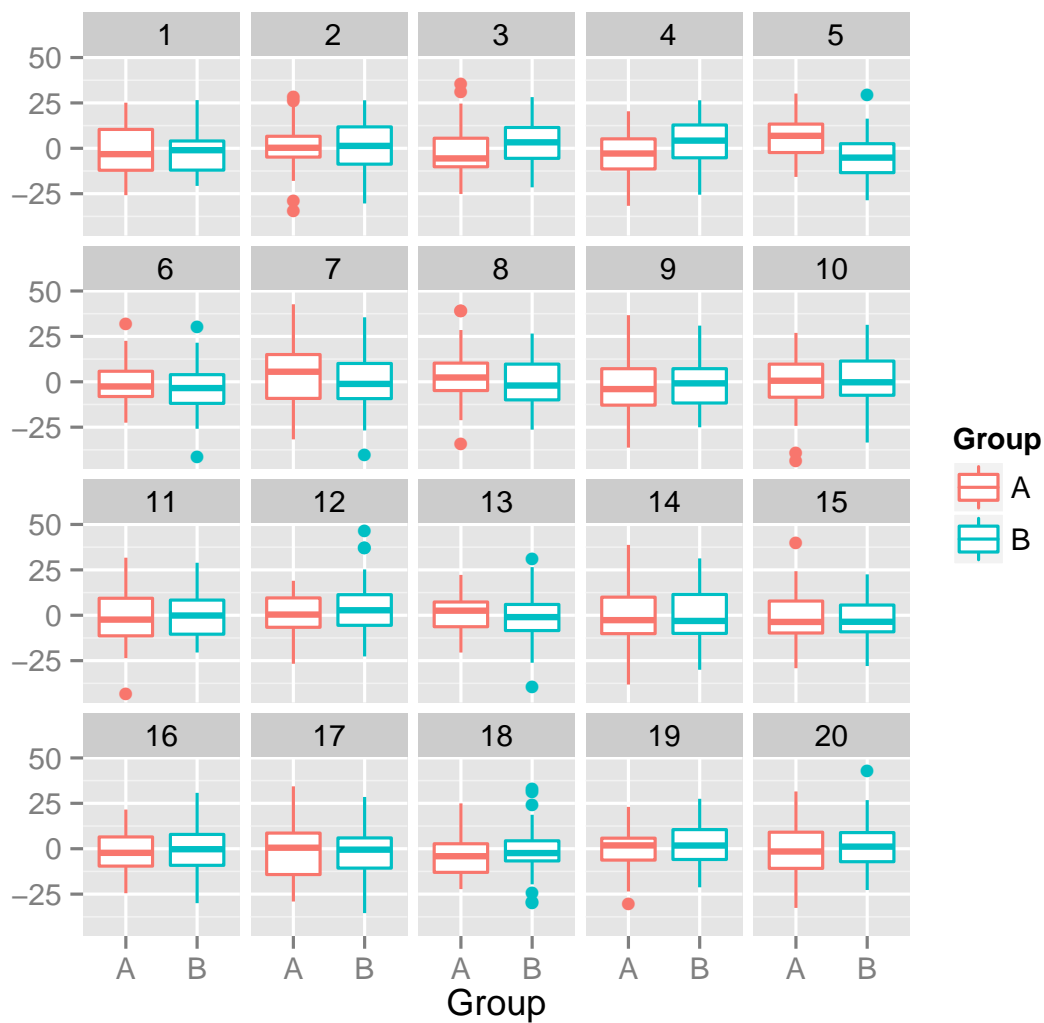
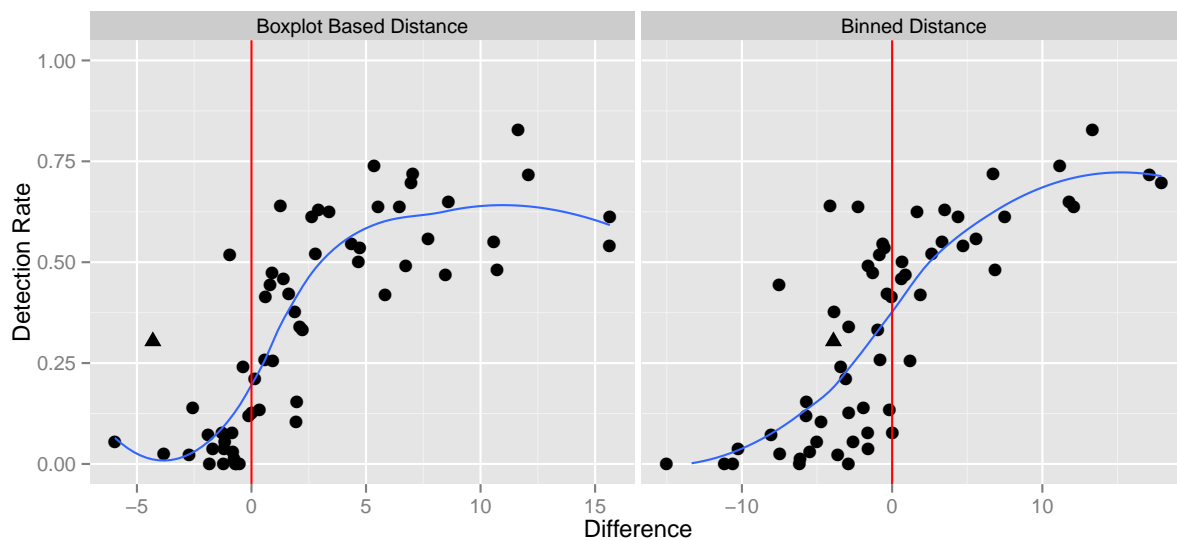
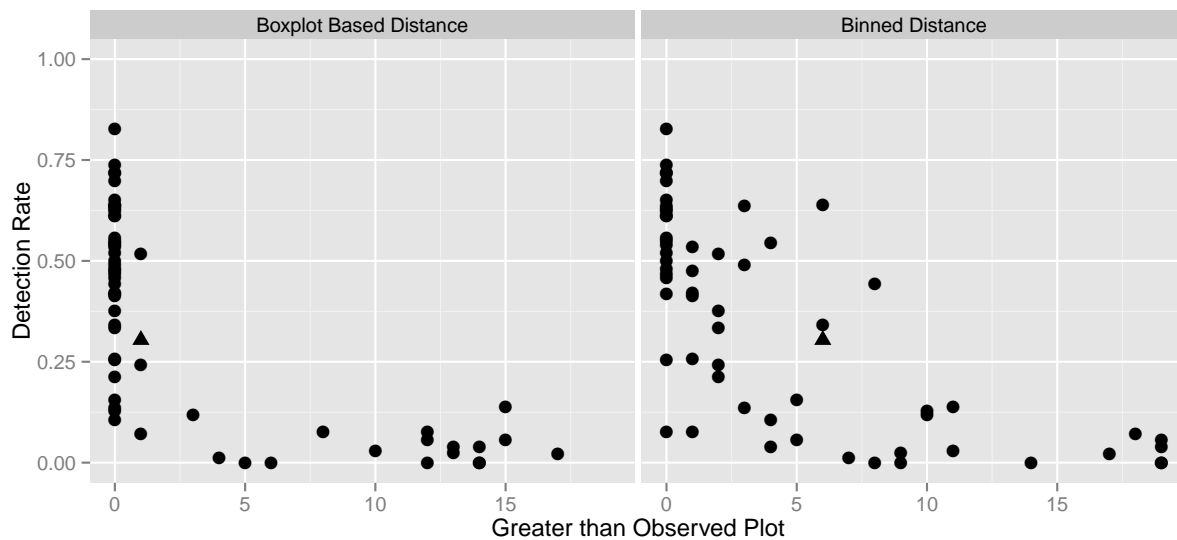


Figure 3.8 An example lineup from Turk Experiment 1. The lineup has $m = 20$ plots of which one is the observed data plot and the remaining $m - 1$ are the null plots generated assuming that the null hypothesis is true. Subjects were asked to identify the plot which has the largest vertical difference between the two groups. Can you identify the observed data plot?



(a)



(b)

Figure 3.9 Comparison of distance metrics for side-by-side boxplots. Detection Rate (a) and the number of plots greater than the observed (b) are plotted against the difference based on the boxplot and binned distance. The vertical line represents the difference equal to 0 when there is at least one null plot similar to the observed plot. The detection rate increases with the difference. As the number of plots with distance greater than the observed increases, the detection rate decreases. The triangle represents a lineup which has high detection rate but negative difference. This particular lineup is examined in Figure 3.11.

look at in the lineup.

The time taken to respond by the subjects is another measure of difficulty of the lineups. Due to the presence of some huge outliers, the mean time taken by the subjects for each lineup is looked at and plotted against the difference for both the distance measures. Figure 3.10 shows the plots. It can be clearly seen that when the difference is below 0, there is no real trend in the mean time and there is a huge variability, indicating that the time taken depends on the subjects. But when the difference is above 0, the mean time decreases rapidly as the difference increases. Hence the subjects can pick the true plot quickly if the true plot is extreme compared to the null plots.

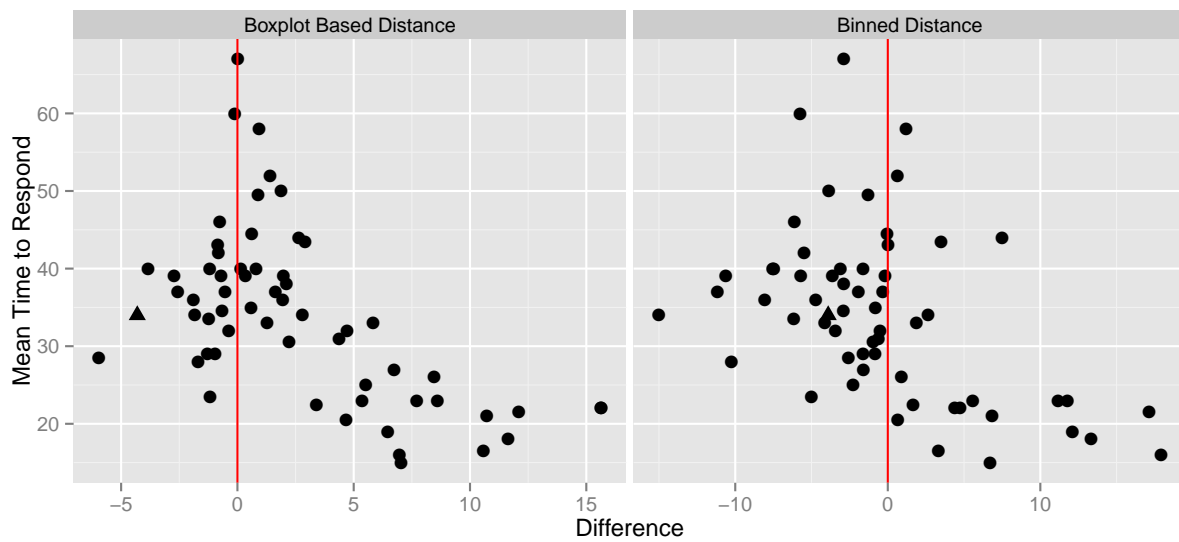


Figure 3.10 Comparison of distance metrics for side-by-side boxplots. Mean time to respond is plotted against the difference based on the boxplot and binned distance. The vertical line represents the difference equal to 0 when there is at least one null plot similar to the observed plot. The mean time taken decreases with the difference. The triangle represents a lineup which is examined in Figure 3.11.

It can be noticed in Figure 3.9 that for some of the lineups, the detection rate is high but the difference using distance metric is negative suggesting that the lineup is difficult. One such lineup is marked using a triangle in Figure 3.9. It would be interesting to look into the lineup closely to identify what made the people pick the actual plot as different. Figure 3.11 shows the lineup and the distribution of the distance metrics.

The lineup in Figure 3.11 is a lineup of side-by-side boxplots. The observed data plot is Plot 20

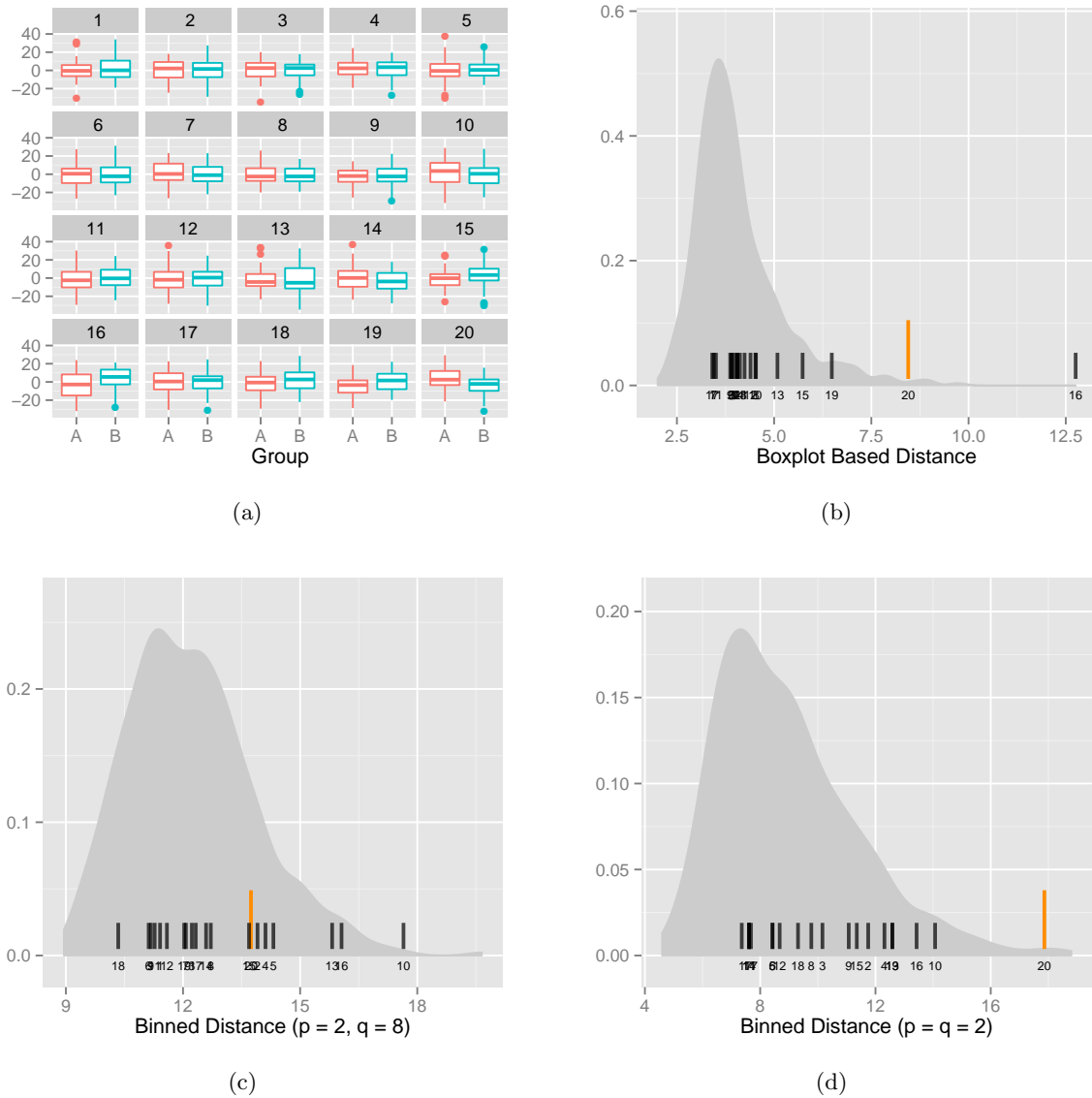


Figure 3.11 Illustration of the behavior of the different distance metrics. The lineup is shown in (a) and the distributions of different distance metrics based on this lineup is shown in the other plots: boxplot based distance in (b), binned distance with 2 and 8 bins on x and y axis in (c) and binned distance with 2 bins in both axes in (d). The lineup corresponds to the point marked with a triangle in difference vs. detection rate plot in Figure 3.9.

but there are other candidates who can be picked easily. Plot 19 and Plot 16 seems to have large differences between the quartiles. Specifically in Plot 16, the difference between the first quartiles for the two groups is very large but the differences between the medians and the third quartiles are small. The huge difference of the first quartiles may have affected the huge mean distance of Plot 16 from all the other plots.

3.8.2 Turk Experiment 2 – Scatterplots with an Overlaid Regression Line

In this experiment, the test statistic is a scatterplot with the regression line overlaid. Assuming that the null hypothesis is true, the null plots are generated by assuming that there is no significant linear relationship between the two variables. The subjects were shown a few lineups and were asked to identify the plot which has the steepest slope. Figure ?? show example lineups.

The distances between the plots in this experiment were computed using both the distance based on regression line (d_{reg}) and the binned distance (d_{bin}) with a small number of bins. The proportion of correct response for each lineup was calculated from the response of the subjects and plotted against δ_{lineup} and γ_{lineup} . Figure 3.13 shows the results for the distance based on the regression line and the binned distance against δ_{lineup} .

Figure 3.13 shows the detection rate against the difference. The vertical line represents difference equal to 0. It can be seen that as the difference increases, the detection rate increases. So the subjects do better in the easier lineups than the hard ones. The distance based on regression works well in capturing the complexity of the lineups. The difference is positive for lineups with large differences and negative for lineups with small difference. A few lineups have difference close to zero for which the detection rate is close to 50%.

But the binned distance fails terribly. Although the detection rate increases with difference, the detection rate is high for values with negative difference. This is a classic scenario where a graphical element affects the response. The presence of the overlaid regression line on almost transparent points of the scatterplot affected the response of the subjects. One other reason may be the use of the same number of bins (2×2) in this case for all the lineups. The relationship may improve if different number of bins can be used for different lineups.

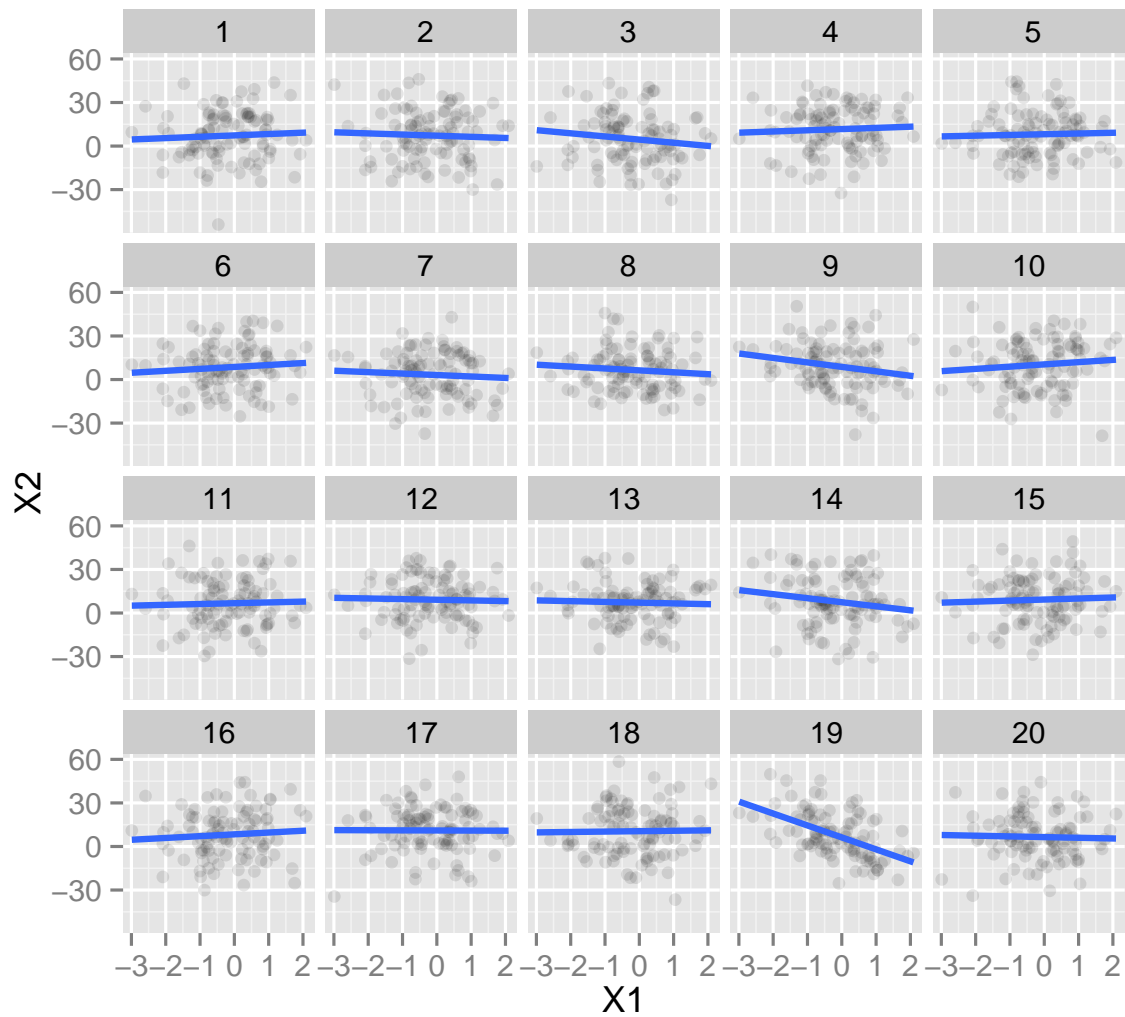
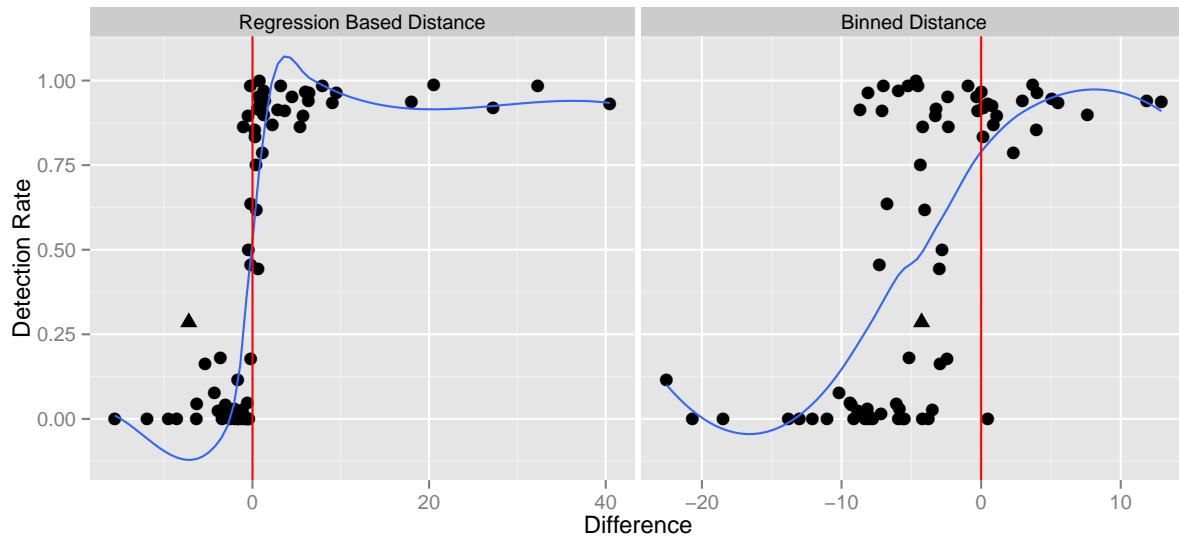
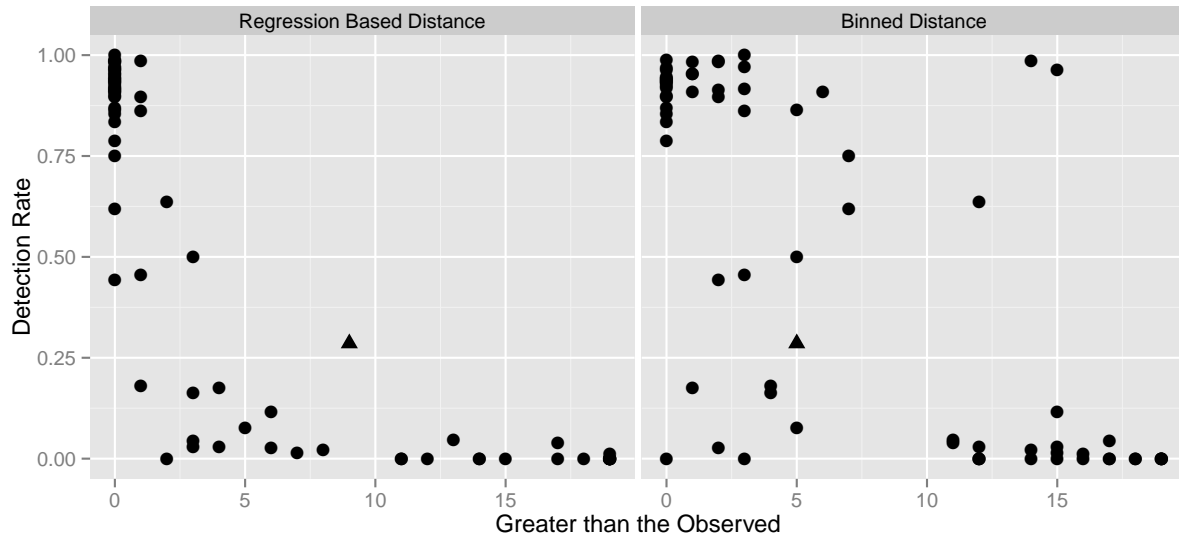


Figure 3.12 An example lineup from Turk Experiment 2. In this lineup, one of the plots is the observed plot and the other 19 plots are the null plots generated assuming that the null hypothesis $H_o : \beta = 0$ is true. Subjects were asked to identify the plot with the steepest slope. Can you identify the observed plot ?



(a)



(b)

Figure 3.13 Comparison of distance metrics for scatterplots with a regression line overlaid. Detection Rate (a) and the number of plots greater than the observed (b) are plotted against the difference based on the regression based and binned distance. The vertical line represents the difference equal to 0 when there is at least one null plot similar to the observed plot. The detection rate increases with the difference. As the number of plots with distance greater than the observed increases, the detection rate decreases. The triangle represents a lineup which has high detection rate but negative difference. This particular lineup is examined in Figure 3.15.

Figure 3.13 also shows the detection rate against the number of null plots greater than the observed plot. As the number of plots greater than the observed plot increases, the detection rate decreases. Hence as there are more extreme null plots compared to the observed plot, the subjects find it difficult to pick the observed plot. For a few lineups, almost all the subjects identify the observed plot although there is one more extreme null plot. Though from Figure 3.13, it can be seen that the extremeness is marginal in most cases.

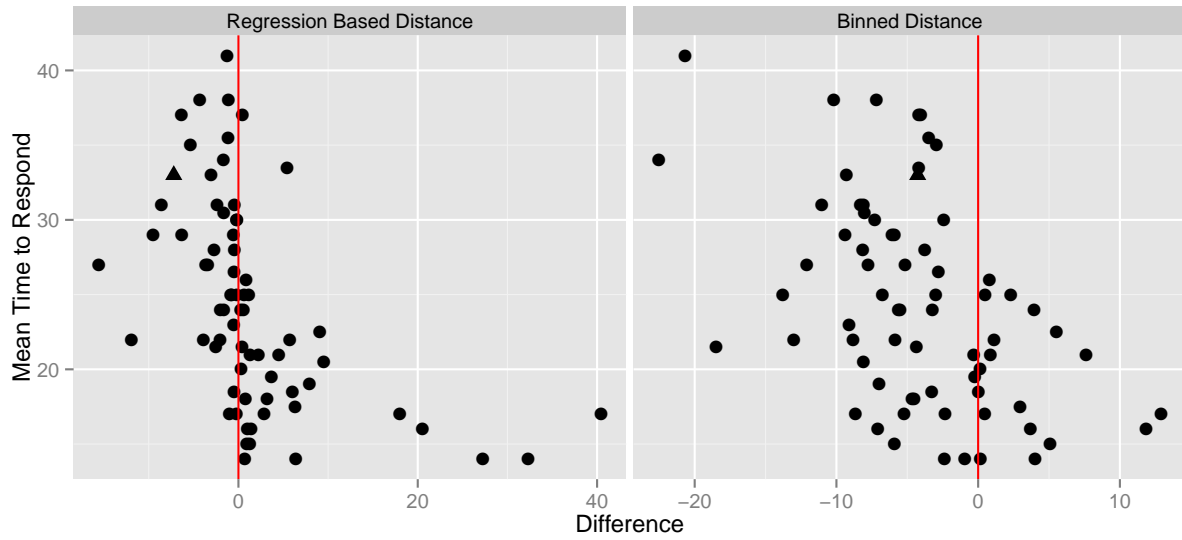


Figure 3.14 Comparison of distance metrics for scatterplots with a regression line over laid. Mean time to respond is plotted against the difference based on the regression based and binned distance. The vertical line represents the difference equal to 0 when there is at least one null plot similar to the observed plot. The mean time taken decreases with the difference. The triangle represents a lineup which is examined in Figure 3.15.

Figure 3.14 shows the relationship between the mean time taken to respond and the difference for both the distances. It can be clearly seen that there is a strong negative association showing that as the difference increases, the subjects take less time to respond. Also the variability of the mean time is higher for smaller difference. In case of binned distance, the relationship is negative though the variability is higher for the above mentioned reasons.

Although the regression based distance seems to efficiently identify the quality of the lineup, there is one lineup (marked by a solid triangle in Figure 3.13) which had a negative difference

although people identified the true plot with reasonable success. Figure 3.15 shows the lineup and the distributions of different distance metrics.

The lineup in Figure 3.15 is a difficult one as suggested by the distribution of the distance metrics based on regression. Although around 28% of the people identified the true plot correctly, the conventional p -value for testing the slope equal to 0 is 0.085, which shows that the relationship is not significant. The binned distance with 2 bins on each axes also shows the same. However the binned distance using the optimal number of bins (8 on the x-axis and 2 on the y-axis) by the optimal number of bins selection method identifies the true plot as different from the others.

3.8.3 Turk Experiment 7 – Large p , Small n Data

The motivation behind this experiment is to study the effect of large dimensions in a data with complete noise and some real separation. Data was simulated with different dimensions and fixed sample size. Data was divided into two or three groups. A projection pursuit with Penalized Discriminant Analysis Index was used and the one and two dimensional projections were obtained. The one or two dimensional projections were then plotted which resulted in the observed data plot. To generate the null data, the group variable in the data was permuted and the projection pursuit was applied. The subjects were shown these lineups and were asked to identify the plot with the most separated colored groups. Figure 3.16 gives an example of such a lineup with two dimensional projections with 3 colored groups.

The distances between the plots in this experiment were computed using the distance based on minimum separation and average separation of the clusters and also the binned distance. The number of bins used for the lineups with one dimensional projections is larger (10 in this case) but for the lineups with two dimensional projections, the number of bins used is 5. The proportion of correct response is plotted against δ_{lineup} and γ_{lineup} for both the distances. Figure 3.17 shows the results.

In Figure 3.17, the detection rate is plotted against the difference for distance based on minimum separation, average separation and the binned distance. The red vertical line shows difference equal to 0. It can be seen that as the difference increases, the detection rate increases and both the

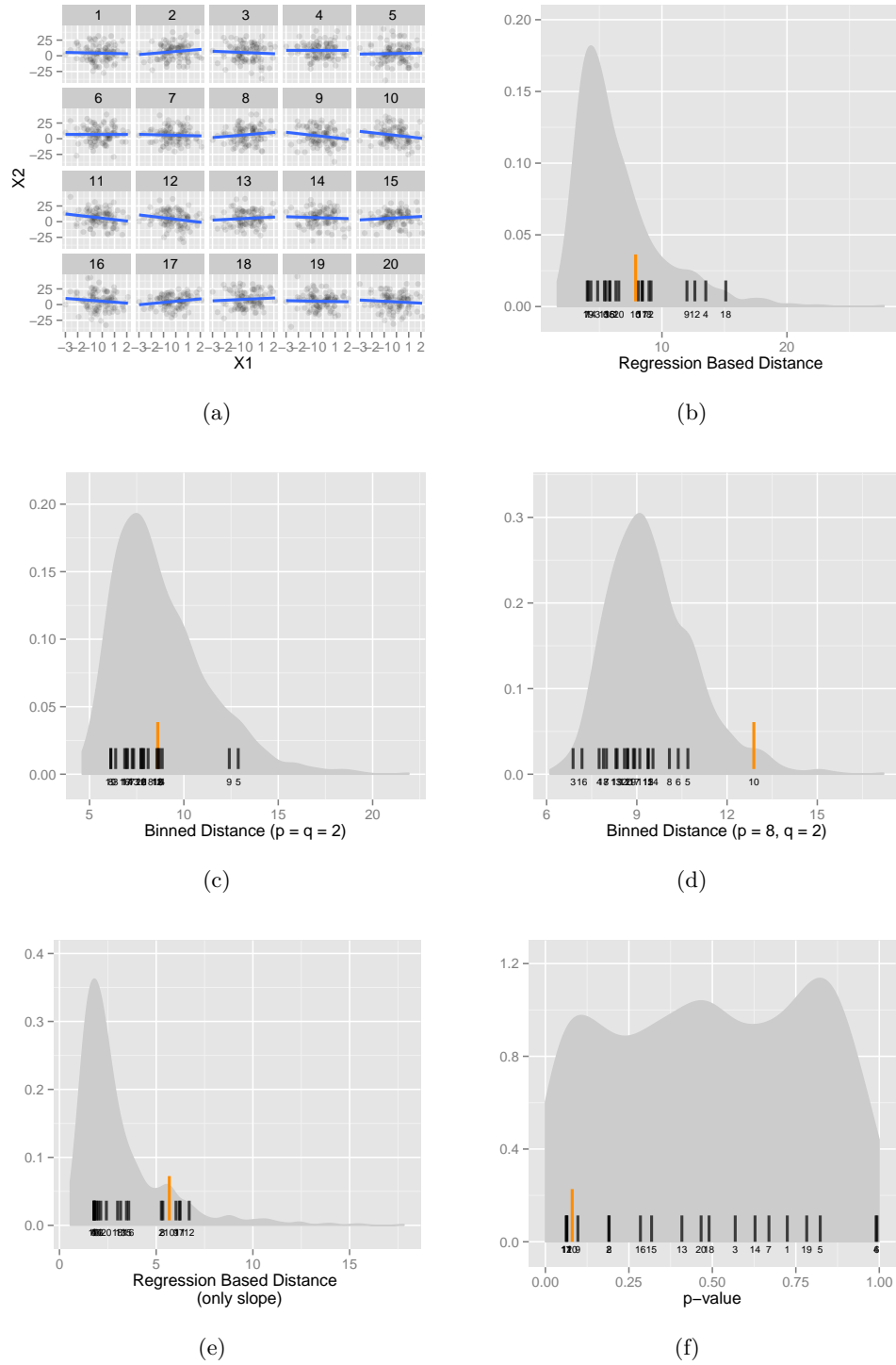


Figure 3.15 Illustration of the behavior of different distance metrics. The lineup is shown in (a) and the distributions of different distance metrics using this lineup is shown in the other plots ((b) - (e)): regression based distance in (b), binned distance with 2 bins on each axes in (c), binned distance with 8 and 2 bins in x and y axis respectively in (d) and regression distance based only on slope in (e). In (f), the distribution of the conventional p -values are plotted with p -values for the lineups marked on the distribution. The lineup corresponds to the point marked with a triangle in difference vs. detection rate plot in Figure 3.13.

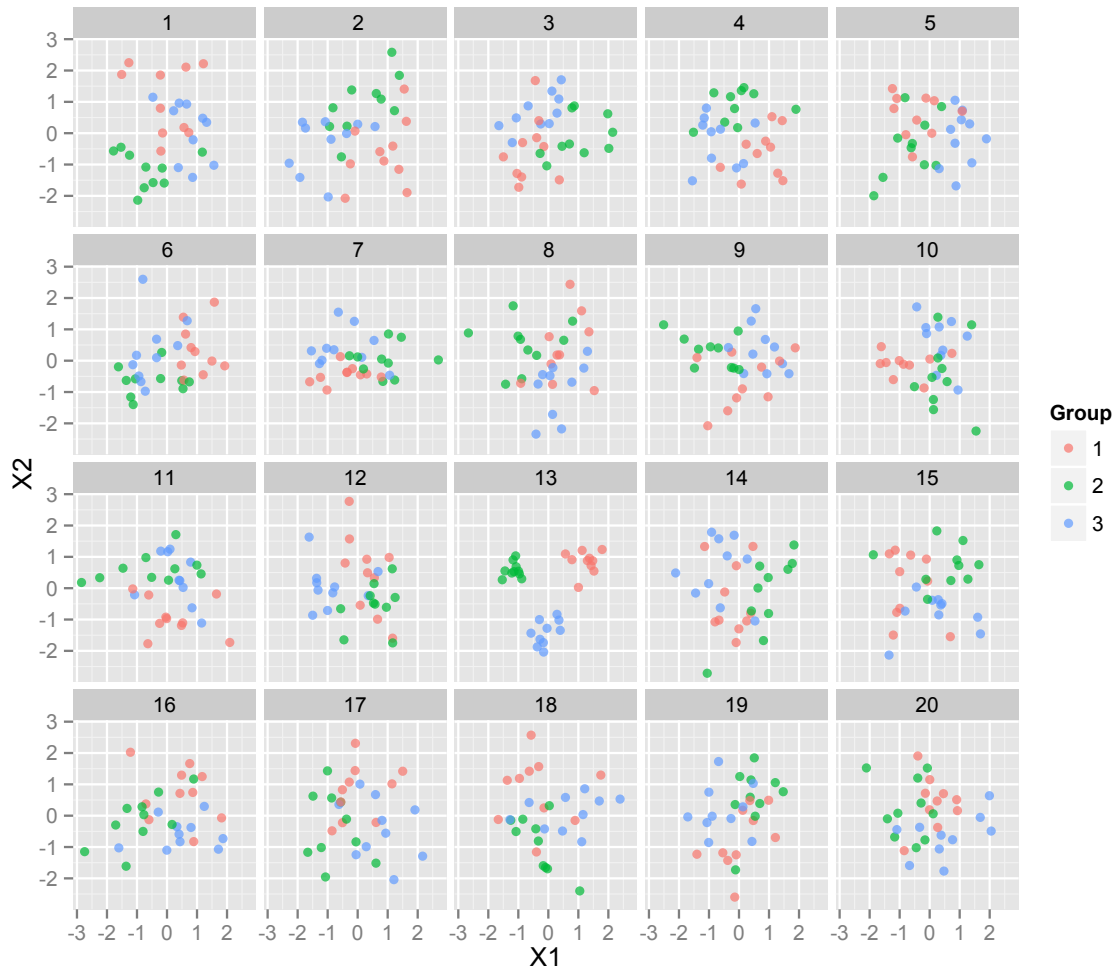
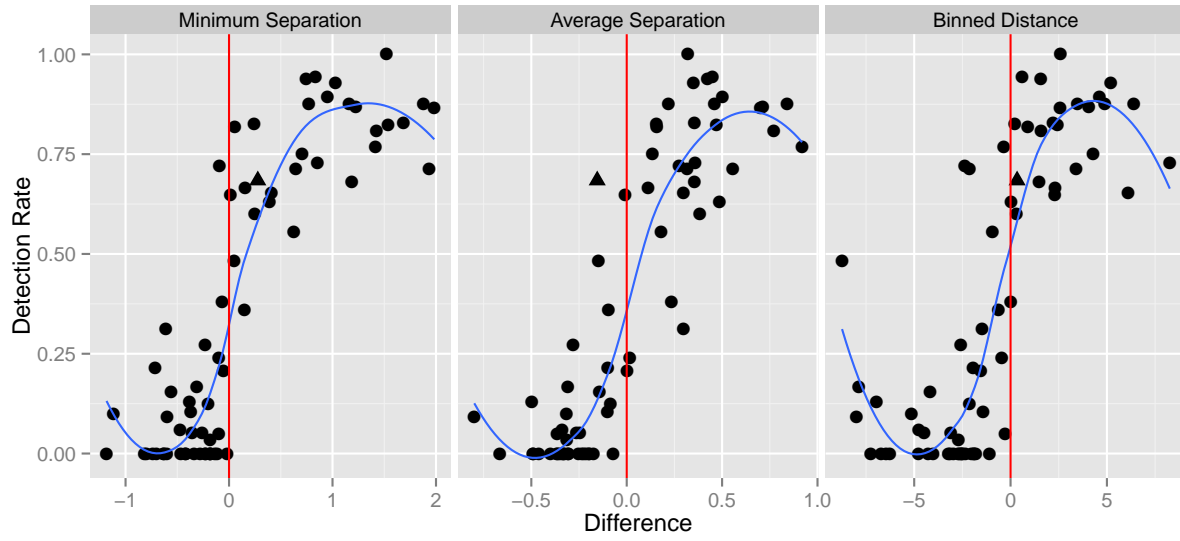
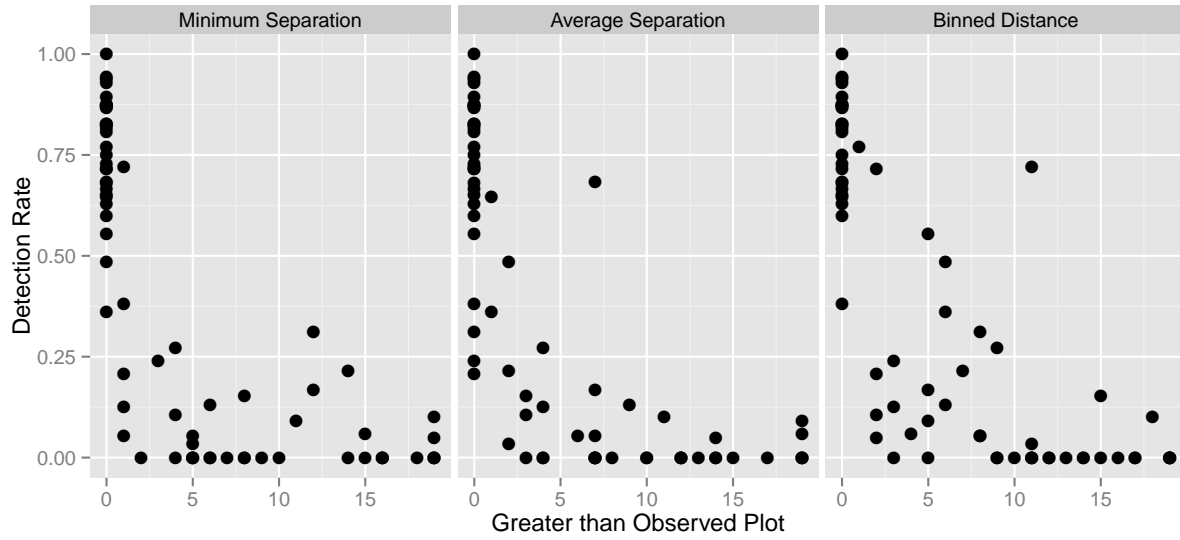


Figure 3.16 An example lineup from Turk Experiment 7. Here two dimensional projections of the PDA index are plotted for a data with separation having $p = 20$ dimensions and $n = 30$ observations. The subjects were asked to identify the plot with the most separated colors. Can you identify the observed data plot?



(a)



(b)

Figure 3.17 Comparison of distance metrics for the scatterplot with clusters. Detection rate (a) and the number of plots greater than the observed (b) are plotted against the difference based on the minimum separation, average separation and binned distance. The vertical line represents the difference equal to 0 when there is at least one null plot similar to the observed plot. The detection rate increases with the difference. As the number of plots with distance greater than the observed increases, the detection rate decreases. The triangle represents a lineup with high detection rate and negative difference based on the average separation distance. This is examined in Figure 3.19.

distances do a good job in capturing the response of the subjects. In (b) it can be seen that as there are more extreme null plots compared to the observed plot, the subjects find it difficult to pick the observed plot. For a few lineups, a large number of the subjects identify the observed plot although there is more extreme null plots.

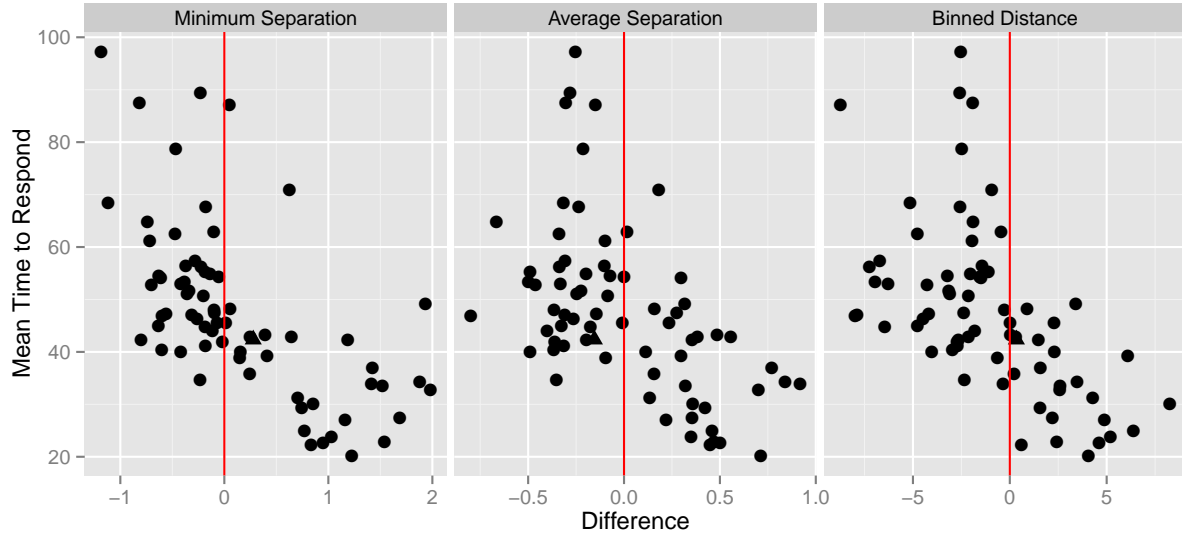
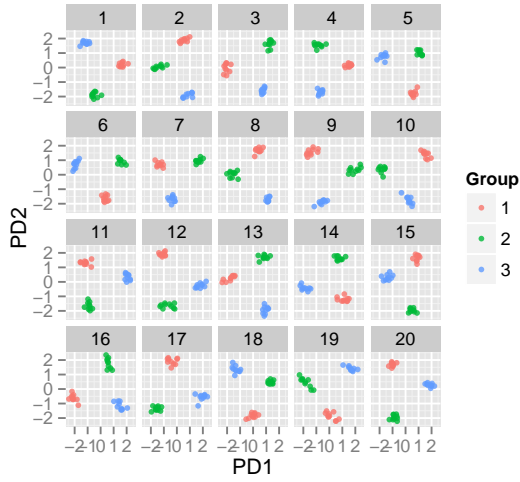


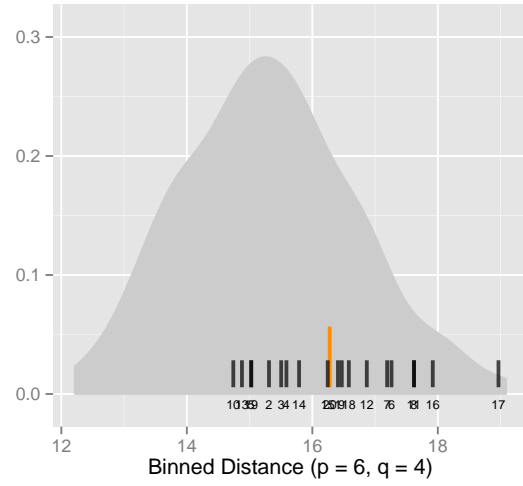
Figure 3.18 Plot showing the mean time to respond by the subjects against the difference based on the minimum separation distance, average separation and binned distance. The vertical line represents the difference equal to 0 when there is at least one null plot similar to the observed plot. The mean time decreases as the difference increases.

Figure 3.18 shows the relationship between the mean time taken to respond and the difference for the three different distances. It can be clearly seen that there is a strong negative association showing that as the difference increases, the subjects take lesser time to respond. Also the variability of the mean time is higher for smaller difference. In case of binned distance, the relationship is negative though the variability is higher for all differences.

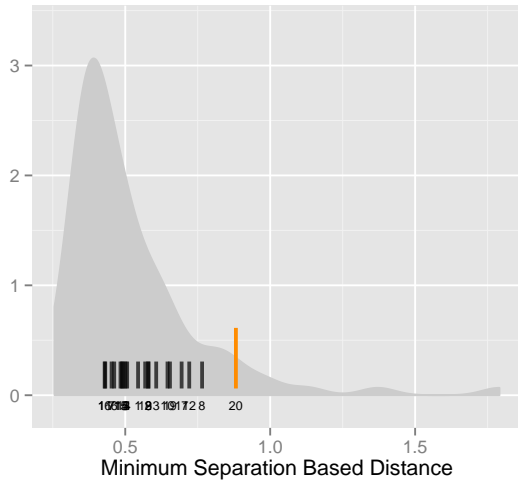
Figure 3.19 shows the lineup in a high dimension, low sample size setting. The number of dimensions used is 100 and two of the dimensions have some separation. Plot 20 shows the two-dimensional projections of the original data. The null plots are obtained by permuting the group variable and plotting the two dimensional projections obtained from a projection pursuit with PDA index (Lee and Cook (2010)). Since the true plot has real separation, it is expected that the subjects



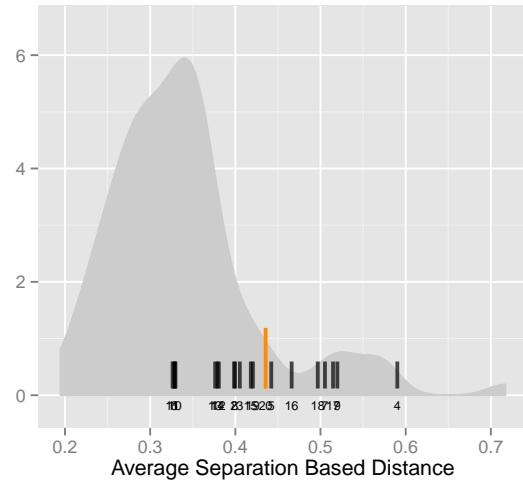
(a)



(b)



(c)



(d)

Figure 3.19 Illustration of the behavior of different distance metrics. The lineup is shown in (a) and the distributions of different distance metrics are shown in the other plots: binned distance with 6 and 4 bins in x and y axis respectively in (b), distance based on average separation in (c) and distance based on minimum separation in (d).

would be able to identify the plot. The distance based on average separation yields a negative difference showing that the lineup is difficult, while the distance based on minimum separation yields a positive difference. The distance metrics identifies different characteristics in a plot. The average separation looks at the average of the distances of the points in a cluster to the points in other clusters. The presence of an outlier point in the opposite side of the other clusters affects this distance considerably. On the other hand, the minimum separation looks at the minimum of the distances. Hence it is not affected by the outlier point.

3.9 Conclusion

Distance metrics are compared to the response of human subjects on lineups. They are comparable to a certain extent except in certain situations where they disagree. There seems to be various reasons behind the disagreement. When people look at a lineup, they may identify a plot as different from the others due to various reasons. But the distance metrics are constructed such that it takes into specific properties of the plot.

Distance metrics can be used to measure the quality of a lineup before showing the lineups to human subjects. Hence the distance metrics allows us to provide a range of lineups to the human subjects to evaluate.

In classical inference, the test statistic under null hypothesis follows a certain distribution. Similarly the null plots in visual inference can also be assumed to be random samples from a sampling distribution. Though theoretically this is true, practically it is impossible to investigate such a distribution. The distribution of the distance metrics approximates such a sampling distribution for a given distance metric. The value of the distance metric for the actual plot can be compared to all the other plots using such a distribution.

The reason of choice can provide a way of evaluating the performance of a distance metric. For example, for a lineup of scatterplots with regression line overlaid, if the choice of reason for majority is steepest slope, the regression based distance may work better than the binned distance. Similarly if the reason of choice is presence of outliers, the binned distance with large number of bins on both

axes may be the best distance metric. This can be a probable future work.

Acknowledgement: This work was funded by National Science Foundation grant DMS 1007697. All plots are done with the `ggplot2` (Wickham, 2009) package in R.

CHAPTER 4. NULLABOR: AN R PACKAGE FOR VISUAL STATISTICAL INFERENCE

A paper to be submitted to *Journal of Statistical Software*.

Niladri Roy Chowdhury, Hadley Wickham, Dianne Cook, Heike Hofmann,
Mahbubul Majumder

Abstract

Statistical graphics play an important role in statistical analysis, model checking and data mining, but they have not been used for statistical inference. Buja et al. (2009) introduced protocols to allow inference to statistical graphics, which is also described in Wickham et al. (2010). Later Majumder et al. (2013) describes a comparative study between classical inference methods and visual inference methods in a linear modeling setting. The recent developments in visual inference demands an open source platform which can be used to implement the visual methods. The R package **nullabor** provides methods of generating lineups automatically for different null generating methods and also construction of diagnostic plots to measure the quality of a lineup. In this paper we discuss the different tools in **nullabor** which helps to use visual inference easily.

Keywords: visual inference; lineup; null generating mechanisms

4.1 Introduction

Recent developments suggest that statistical graphics can be used to perform statistical inference. Buja et al. (2009) introduced protocols to allow inference to statistical graphics, which were later validated by Majumder et al. (2013). The recent developments require a software package which would make the implementation of visual inference easier and automatic. Buja et al. (2009) describes two protocols which are known as the lineup protocol and the Rorschach protocol, which is also described in Wickham et al. (2010). In the lineup protocol, the plot of the actual data is randomly placed in a set of null plots. These null plots are generated by a mechanism which is consistent with the null hypothesis. A human judge is shown the lineup and asked to identify the plot which is the most different or the plot which has the strongest structure. If the human judge can identify the plot of the actual data, we reject the null hypothesis. On the other hand, in the Rorschach protocol, all the plots are generated from the null distribution assuming that the null hypothesis is true. This protocol helps us to calibrate our eyes to the natural variability of the plots and hence reduce our sensitivity to structures which appear purely due to randomness. This package provides functions to implement these two protocols in different scenarios.

The second part of the package introduces distance metrics. Distance metrics are used to measure distances between the plots in a lineup. The null plots in a lineup may affect the decision of the human subjects since they base their decision on a finite number of null plots. Hence it is essential to measure the quality of the lineups by calculating the distances between the plot of the actual data and the null plots and the null plots among themselves. To compare several lineups, the distances obtained from a certain lineup are compared to the empirical distribution of the distances.

This chapter is organized in the following way. Section 4.2 describes the different null generating mechanisms which are used to obtain the null plots in the lineup. The two protocols are described in Section 4.3. Section 4.4 discusses five different distance metrics. These are targeted for different varieties of data and plot types. Calculating different statistics from a lineup are described in the next section. Section 4.6 describes the procedure to select the optimal number of bins for binned distance. Finally Section 4.7 gives the method of obtaining and plotting the empirical distribution

of the distance metrics.

4.2 Null generating mechanisms

The null plots in a lineup is obtained by a method which is consistent with the null hypothesis. Assuming that the null hypothesis is true, the null datasets are obtained, often, from the actual dataset. This process of generating the null datasets is called the null generating mechanism. There may be a variety of null generating mechanisms but the package provides three different methods.

4.2.1 Generate null data with a specific distribution

The `"null_dist"` function takes as input a variable name of the data and a particular distribution. This variable in the data is substituted by random generations of the particular distribution. The different distributions include beta, cauchy, chi-squared, exponential, f, gamma, geometric, log-normal, lognormal, logistic, negative binomial, normal, poisson, t and weibull. A list of parameters of distribution can also be provided as input. In case it is not provided, `"fitdistr"` is used to estimate the parameters from the given data. The function `"null_dist"` returns a function that given the data generates a null data set.

```
> head(null_dist("mpg", dist = "normal")(mtcars))
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	29.536386	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	16.897772	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	16.395593	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	9.644472	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	11.456224	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	25.417741	6	225	105	2.76	3.460	20.22	1	0	3	1

4.2.2 Generate null data by permuting a variable

The "null_permute" function takes as input a variable name of the data. This variable is permuted to obtain the null dataset. The function "null_dist" returns a function that given the data generates a null data set.

```
> head(null_permute("mpg")(mtcars))
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	19.7	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.4	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	30.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	21.0	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	15.8	6	225	105	2.76	3.460	20.22	1	0	3	1

4.2.3 Generate null data with null residuals from a model

The function "null_lm" takes as input a model specification formula as defined by "lm" and method for generating null residuals from the model. The three built in methods are 'rotate', 'pboot' and 'boot' defined by "resid_rotate", "resid_pboot" and "resid_boot" respectively. The function returns a function which given the data generates a null dataset.

```
> head(null_lm(wt~mpg, method = 'rotate')(mtcars))
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	3.049169	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.634875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.917568	18.61	1	1	4	1

Hornet 4 Drive	21.4	6	258	110	3.08	3.126533	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	4.281755	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.368690	20.22	1	0	3	1
		.resid			.fitted						
Mazda RX4	-0.03998504				3.089154						
Mazda RX4 Wag	-0.45427860				3.089154						
Datsun 710	0.08196630				2.835602						
Hornet 4 Drive	0.09372421				3.032809						
Hornet Sportabout	0.86861876				3.413136						
Valiant	-0.12896315				3.497653						

4.3 Protocols

Buja et al. (2009) introduces two protocols which bridges the gulf between traditional statistical inference and exploratory data analysis. The two protocols can be implemented in the following way:

4.3.1 The lineup protocol

In this protocol, the plot of the real data is randomly embedded amongst a set of null plots. The matrix of plots is known as a lineup. The null plots are generated by a method consistent with the null hypothesis. The lineup is shown to an observer. If the observer can pick the real data as different from the others, this puts weight on the statistical significance of the structure in the plot. The **"lineup"** function returns a set of generated null datasets and the real data embedded randomly among these null datasets. The method of null generation should be provided in the lineup function for the null datasets to be generated automatically along with the real dataset. The users also have the option of generating the null datasets themselves and providing them in the **"lineup"** function. The position of the real dataset can be left missing and the function picks the position at random. The function then returns the position as an encrypted code. The encrypted

code is copied and pasted on the R console to obtain the true position of the plot.

```
> head(lineup(null_permute("mpg"), mtcars), 20)
```

```
decrypt("jriG PHhH D5 QZXDhDZ5 Jf")
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb	.sample
1	15.8	6	160.0	110	3.90	2.620	16.46	0	1	4	4	1
2	16.4	6	160.0	110	3.90	2.875	17.02	0	1	4	4	1
3	21.4	4	108.0	93	3.85	2.320	18.61	1	1	4	1	1
4	30.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1	1
5	17.8	8	360.0	175	3.15	3.440	17.02	0	0	3	2	1
6	18.7	6	225.0	105	2.76	3.460	20.22	1	0	3	1	1
7	19.2	8	360.0	245	3.21	3.570	15.84	0	0	3	4	1
8	10.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2	1
9	21.0	4	140.8	95	3.92	3.150	22.90	1	0	4	2	1
10	27.3	6	167.6	123	3.92	3.440	18.30	1	0	4	4	1
11	15.5	6	167.6	123	3.92	3.440	18.90	1	0	4	4	1
12	15.2	8	275.8	180	3.07	4.070	17.40	0	0	3	3	1
13	18.1	8	275.8	180	3.07	3.730	17.60	0	0	3	3	1
14	21.0	8	275.8	180	3.07	3.780	18.00	0	0	3	3	1
15	15.2	8	472.0	205	2.93	5.250	17.98	0	0	3	4	1
16	22.8	8	460.0	215	3.00	5.424	17.82	0	0	3	4	1
17	14.3	8	440.0	230	3.23	5.345	17.42	0	0	3	4	1
18	21.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1	1
19	10.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2	1
20	21.5	4	71.1	65	4.22	1.835	19.90	1	1	4	1	1

The lineup data can be then used to generate the lineup using **ggplot2**. The lineup is shown to one or more observers who are asked to identify the plot which is different. If the observer can identify the plot of the real data correctly, we reject the null hypothesis and conclude that the plot of the real data has stronger structure than the null plots.

```
> qplot(mpg, wt, data = lineup(null_permute("mpg"), mtcars)) + facet_wrap(~ .sample)

decrypt("jriG PHhH D5 QZXDhDZ5 1")
```

Figure 4.1 is an example of a lineup plot where the plot of the actual data is embedded into nineteen null plots. The null plots are obtained by permuting the variable “mpg” while keeping the other variables fixed. The position of the actual plot is not known to the user. It can be obtained by copying the output in the R console. Hence the user can also act as a human judge.

4.3.2 The Rorschach protocol

The Rorschach protocol is used to calibrate the eyes for variation due to sampling. The plots generated corresponds to the null datasets, data that is consistent with a null hypothesis. The "rorschach" function returns a set of null plots which are shown to observers to calibrate their eyes with variation. Like the "lineup" function, the null generating mechanism should be provided as an input along with a real dataset. A probability can also be given as input which dictates the chance of including the true data with null data.

```
> head(rorschach(null_permute("mpg"), mtcars, n = 8, p = 0), 20)
```

	.n	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
1	1	17.3	6	160.0	110	3.90	2.620	16.46	0	1	4	4
2	1	32.4	6	160.0	110	3.90	2.875	17.02	0	1	4	4
3	1	21.0	4	108.0	93	3.85	2.320	18.61	1	1	4	1
4	1	21.0	6	258.0	110	3.08	3.215	19.44	1	0	3	1

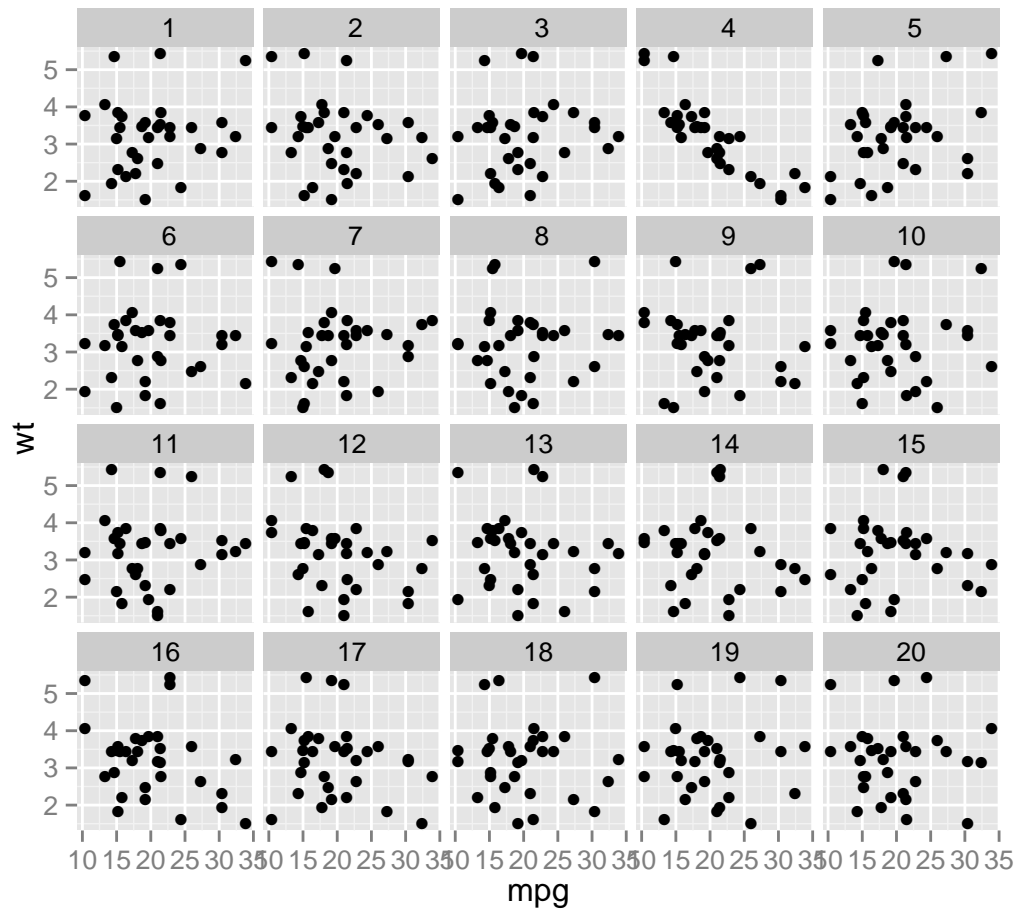


Figure 4.1 A typical lineup. The plot of the actual data is embedded in a set of null plots which are generated assuming that the null hypothesis is true. The variable “mpg” is permuted 19 times to obtain the 19 null plots. Can you identify the plot which has the steepest slope? The true position of the plot can be obtain by copying and pasting “decrypt(…)” from the output into R console.

5	1	15.5	8	360.0	175	3.15	3.440	17.02	0	0	3	2
6	1	30.4	6	225.0	105	2.76	3.460	20.22	1	0	3	1
7	1	14.7	8	360.0	245	3.21	3.570	15.84	0	0	3	4
8	1	22.8	4	146.7	62	3.69	3.190	20.00	1	0	4	2
9	1	15.0	4	140.8	95	3.92	3.150	22.90	1	0	4	2
10	1	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
11	1	22.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
12	1	18.7	8	275.8	180	3.07	4.070	17.40	0	0	3	3
13	1	26.0	8	275.8	180	3.07	3.730	17.60	0	0	3	3
14	1	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
15	1	21.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
16	1	24.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
17	1	21.5	8	440.0	230	3.23	5.345	17.42	0	0	3	4
18	1	19.7	4	78.7	66	4.08	2.200	19.47	1	1	4	1
19	1	15.2	4	75.7	52	4.93	1.615	18.52	1	1	4	2
20	1	21.4	4	71.1	65	4.22	1.835	19.90	1	1	4	1

```
> qqplot(mpg, wt, data = rorschach(null_permute("mpg"), mtcars, n = 9, p = 0))
+ facet_wrap(~ .n)
```

The Rorschach protocol helps us to calibrate our vision to the natural variability in plots in which the data is generated from scenarios consistent with the null hypothesis. Figure 4.2 is drawn using the rorschach data using package **ggplot2**. Can we identify any interesting pattern in the plots? Is there a plot among these which has a stronger structure than others?

4.4 Distance metrics

There are five different distance metrics in **nullabor** package, named "bin_dist", "box_dist", "reg_dist", "sep_dist" and "uni_dist". The different distance metrics are constructed so that they can identify the different properties of the data. "uni_dist" works for univariate data while

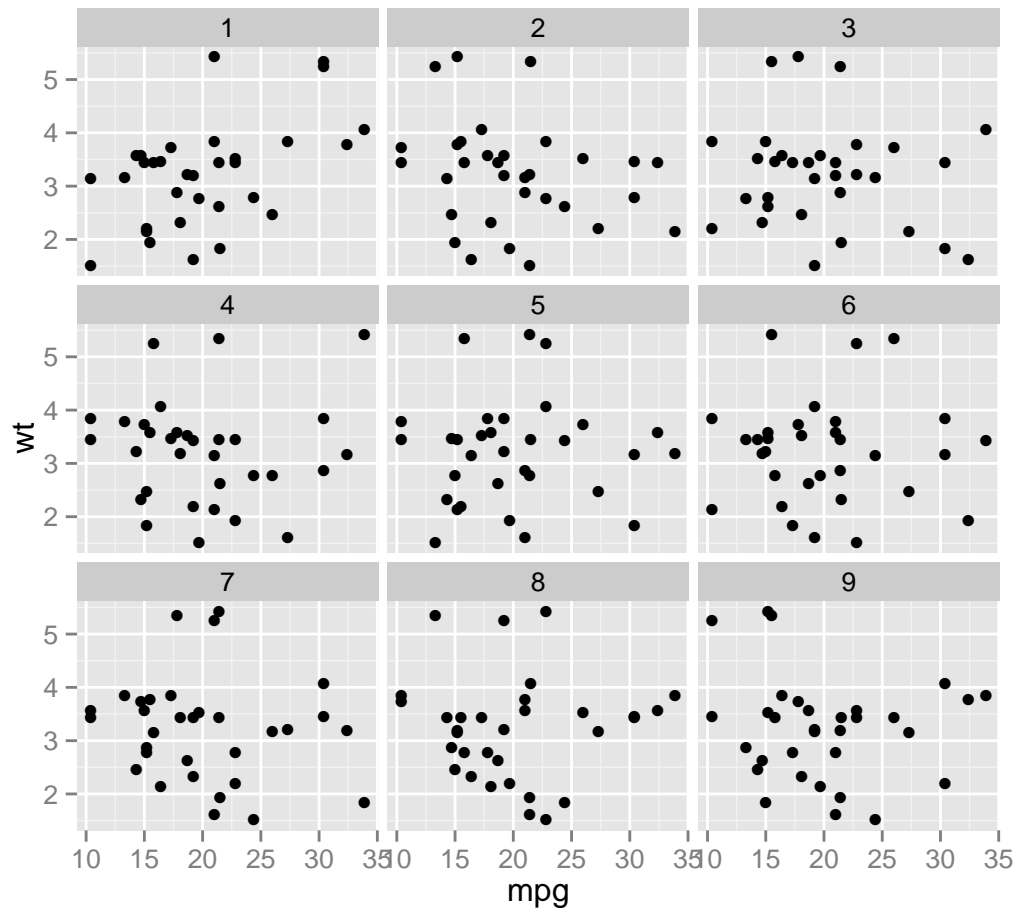


Figure 4.2 Illustration of the Rorschach protocol. These nine scatterplots are obtained by permuting the variable “mpg” while keeping the variable “wt” fixed. Does it look like the strength of the relationship is same in all the plots? Is there a plot which has more interesting pattern than others?

the others works for all types of bivariate data. Binned distance is a generic distance which can be used in any situations while the other distance metrics are constructed so that they can identify the effect of graphical elements in a plot like an overlaid regression line or presence of defined clusters. To calculate some of the metrics, additional informations like a class variable or the number of bins should be provided.

4.4.1 Distance for univariate data

"uni_dist" is a distance metric which calculates the euclidean distance between the first four central moments of two univariate data. A typical usage would be when one needs to calculate the distance between the two histograms drawn from two datasets.

```
> uni_dist(X = rnorm(100), PX = rpois(100, 2))
```

```
[1] 2.980522
```

4.4.2 Distance based on regression parameters

"reg_dist" is a distance metric which calculates the euclidean distance between the regression parameters of a model fitted to one plot and that of another plot. It is advisable to use this distance in situations where a regression line is overlaid on a scatterplot.

```
> X <- data.frame(X1 = mtcars$wt, X2 = mtcars$mpg)
> PX <- data.frame(X1 = sample(mtcars$wt), X2 = mtcars$mpg)
> reg_dist(X, PX)
```

```
[1] 132.9617
```

4.4.3 Distance based on boxplots

"box_dist" is a distance metric which works for side-by-side boxplots with two levels. The first quartile, median and the third quartile are calculated for each box and the absolute distances of

these are calculated for the two boxes. "box_dist" calculates the euclidean distance between these absolute distances for the two plots. The boxplot distance should be used in situations where a side-by-side boxplot is used to compare the distribution of a variable at two different levels.

```
> X <- data.frame(X1 = as.factor(mtcars$am), X2 = mtcars$mpg)
> PX <- data.frame(X1 = as.factor(sample(mtcars$am)), X2 = mtcars$mpg)
> box_dist(X, PX)
```

```
[1] 12.94846
```

4.4.4 Distance based on separation

"sep_dist" is a distance metric based on the separation between clusters. The separation between clusters is defined by the minimum distances of a point in the cluster to a point in another cluster. The separation between the clusters for a given dataset is calculated. An euclidean distance is calculated between the separation for the given dataset and another dataset. The number of clusters in the dataset should be provided. If not, the hierarchical clustering method is used to obtain the clusters.

```
> X <- data.frame(X1 = mtcars$wt, X2 = mtcars$mpg,
  cl = as.numeric(as.factor(mtcars$cyl)))
> PX <- data.frame(X1 = sample(mtcars$wt), X2 = mtcars$mpg,
  cl = as.numeric(as.factor(mtcars$cyl)))
> sep_dist(X, PX, clustering = TRUE)
```

```
[1] 0.2552364
```

```
> sep_dist(X, PX, clustering = FALSE, nclust = 3)
```

```
[1] 0.04571194
```

4.4.5 Binned Distance

"bin_dist" is a generic distance which works for any situation for any dataset. For a given bivariate dataset, X and Y variables are divided into p and q bins respectively to obtain pq cells. The number of points falling in each cell are counted for a given dataset. "bin_dist" between two datasets calculates the euclidean distance between the cell counts of these two data. The values of p and q should be provided as arguments.

```
> X <- data.frame(X1 = mtcars$wt, X2 = mtcars$mpg)
> PX <- data.frame(X1 = sample(mtcars$wt), X2 = mtcars$mpg)
> bin_dist(X,PX, lineup.dat = NULL, X.bin = 5, Y.bin = 5)

[1] 7.745967
```

4.5 Calculation of mean distances and difference

If the plot of the actual data can be easily identified in a lineup, then the plot should have stronger structures than the null plots. The distance metrics can be used to calculate the distance between the plot of the actual data and the null plots. The distances between the null plots can also be calculated.

4.5.1 Calculating the mean distances for the plots in the lineup

The distances between the true plot and all the null plots are calculated and the mean of these distances is calculated. Similarly, for each null plot, the distances between the null plots are calculated and averaged to obtain the mean distance for each null plot. "calc_mean_dist" calculates the mean distance corresponding to each plot in the lineup. If the mean distance of the true plot is larger than the mean distances of all the null plots, the lineup is considered easy. If one of the null plots has a larger mean distance than the true plot, the lineup is considered difficult.

```
> lineup.dat <- lineup(null_permute('mpg'), mtcars, pos = 10)
```

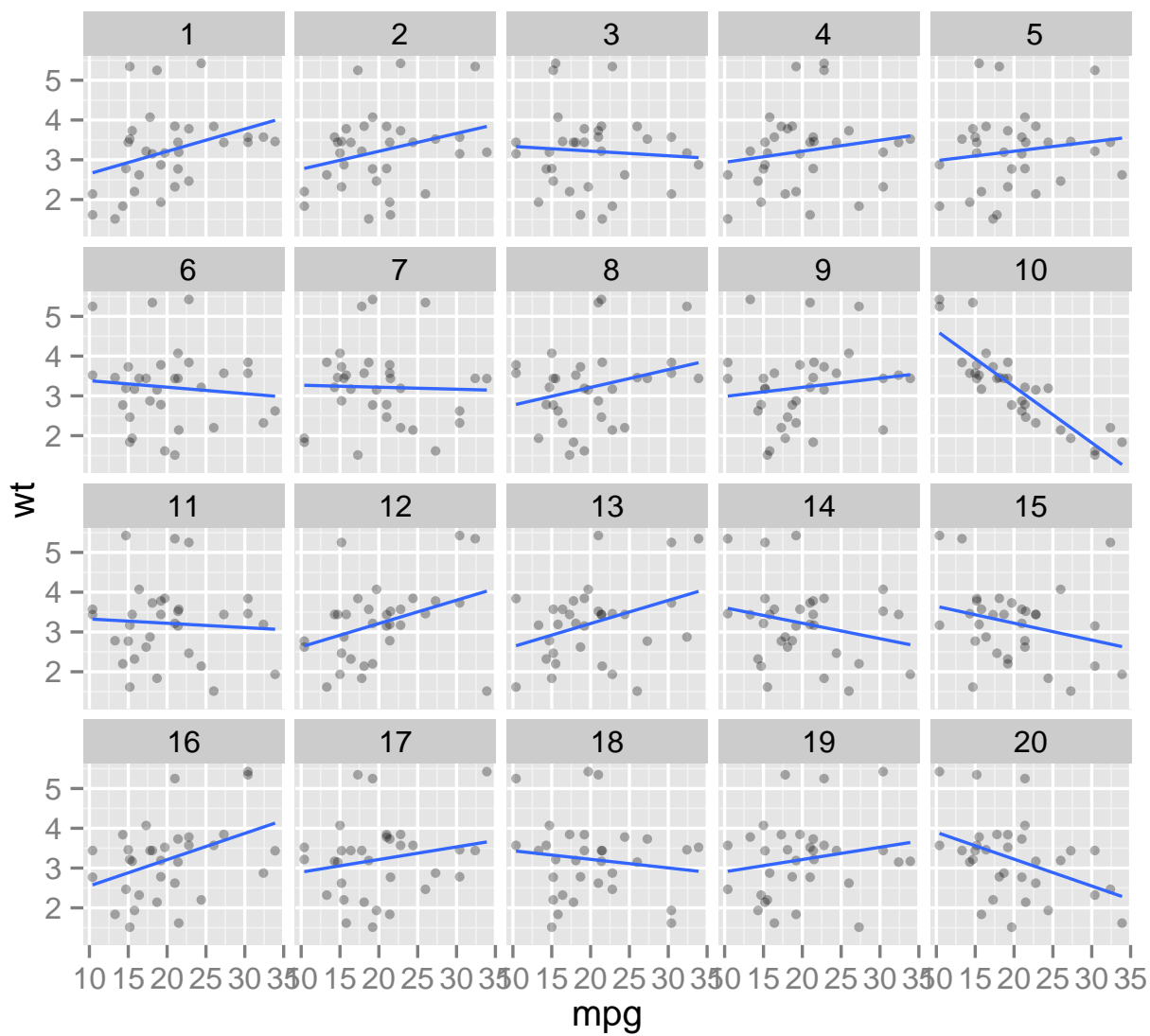


Figure 4.3 Lineup plot of size $m = 20$ of scatterplots between “mpg” and “wt” with a regression line overlaid. The plot of the actual data is randomly placed among the set of null plots. The position of the plot of the actual data is 10.


```
> calc_mean_dist(lineup.dat, var = c('mpg', 'wt'), met = 'reg_dist', pos = 10)
Source: local data frame [20 x 2]
```

	plotno	mean.dist
1	1	1.4105848
2	2	1.0551943
3	3	0.8912037
4	4	0.7146171
5	5	0.6708689
6	6	0.9966672
7	7	0.7674444
8	8	1.0456351
9	9	0.6642878
10	10	10.1982236
11	11	0.8703412
12	12	1.5049771
13	13	1.4878584
14	14	1.7874604
15	15	1.9487011
16	16	1.8179699
17	17	0.7709551
18	18	1.1412758
19	19	0.7568216
20	20	3.4081131

4.5.2 Calculating difference measure for lineups

The mean distances for each plot in the lineup are obtained using `"calc_mean_dist"`. `"calc_diff"` calculates the difference between the mean distance for the true plot and the maximum mean distance

for the null plots.

```
> calc_diff(lineup.dat, var = c('mpg', 'wt'), met = 'reg_dist', dist.arg = NULL,  
pos = 10)
```

```
[1] 6.79011
```

4.6 Selection of bins

Binned distance is highly affected by the choice of the number of bins. The number of bins is provided by the user and this can be subjective. This motivates to design a way to select the optimum number of bins to be used. "opt_diff" finds the optimal number of bins in both x and y direction which should be used to calculate the binned distance. The binned distance is calculated for each combination of provided choices of number of bins in x and y direction and finds the difference using "calc_diff" for each combination. The combination for which the difference is maximum should be used.

```
> opt.diff <- opt_bin_diff(lineup.dat, var = c('mpg', 'wt'), 2, 10, 2, 10, pos = 10,  
plot = TRUE)  
> head(opt.diff$dat)
```

Source: local data frame [6 x 3]

Groups: xbins

	xbins	ybins	Diff
1	2	2	-0.1033782
2	2	3	-0.7769468
3	2	4	-0.4836467
4	2	5	-0.4836467
5	2	6	-0.4956530

```
6      2      7 -0.4836467
```

```
> opt.diff$p
```

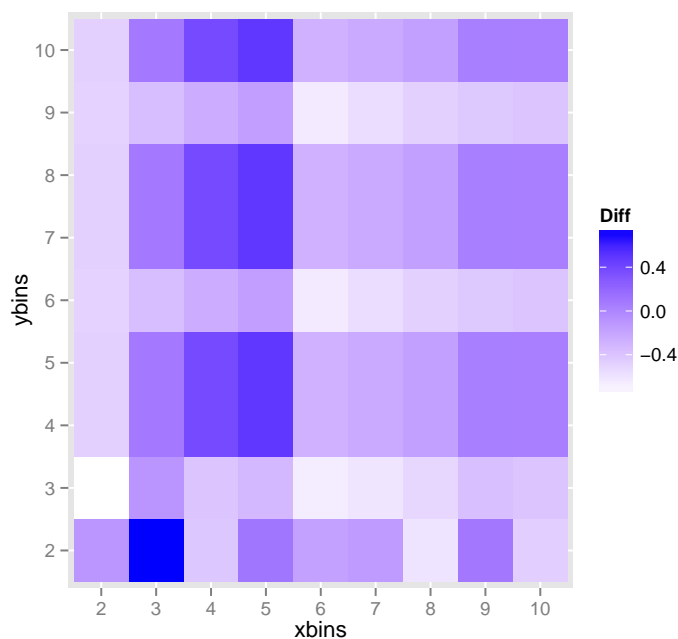


Figure 4.4 Illustration of the optimum bin selection procedure. The obtained lineup data from subsection 4.5.1 is used to generate difference between the true plot and the maximum of the null plots using a binned distance for all combinations of number of bins on x and y axes. The number of bins on x and y axes are plotted on x and y respectively with the colors showing the differences. The darker color represents larger difference.

4.7 Distribution of distance metrics

Comparing several lineups require a common platform. For a given lineup, the distance for the plot of the actual data can be compared to the empirical distribution of all the null plots. An extreme value of this distance indicates that the plot of the actual data is not consistent with the null hypothesis. Hence such a lineup would be considered “easy”. If the same plot of actual data is used with another set of null plots in another lineup, there may be a null plot which is more extreme

than the actual plot. Hence such a lineup would be considered “difficult”. The following functions are used to obtain the empirical distribution of the distance metric and plot the distribution using **ggplot2**.

4.7.1 Empirical distribution of distance metrics

"**distmet**" function provides the empirical distribution of the distance metrics based on the mean distance of the true plot and the mean distance from the null plots. The lineup data, the null generating mechanism and the choice of the distance metric has to be provided. Users have the flexibility of using their distance metrics. The position of the true plot in the lineup has to be provided as well. If the distance metrics require additional arguments, those have to be provided as well.

The distribution of the regression based distance is obtained as following. The output calculates the distances for the plots in the lineup and also the difference between the plot of the actual data and the maximum of the null plots. It also provides the null plots which are closest to the plot of the actual data. The position of the actual data plot and the values for the null distributions are also given to be used for the plotting function.

```
> dist.vals1 <- distmet(lineup.dat, var = c('mpg', 'wt'),'reg_dist', null_permute('mpg'),  
pos = 10, repl = 1000, dist.arg = NULL)  
> head(dist.vals1$lineup)
```

Source: local data frame [6 x 2]

	plotno	mean.dist
1	1	1.4105848
2	2	1.0551943
3	3	0.8912037
4	4	0.7146171

```
5      5 0.6708689
6      6 0.9966672
```

```
> dist.vals1$diff
```

```
[1] 6.79011
```

```
> head(dist.vals1$closest)
```

```
[1] 20 15 16 14 12
```

```
> head(dist.vals1$null_values)
```

```
[1] 1.0564997 0.4444821 0.3020866 0.3106895 0.8703144 0.9713040
```

```
> dist.vals$pos
```

```
[1] 10
```

4.7.2 Plotting the empirical distribution of the distance metric

"distplot" functions plots the empirical distribution of the distance metric, given the output of "distmet" function. The distribution is shown in grey along the distance for the true plot in orange and the distances for the null plots in black.

The empirical distribution of the regression based distance metric is plotted in the following way. This plot can be matched with Figure 4.3. Plot 10 is the plot of the actual data which can easily be identified from the lineup and it can also be seen in Figure 4.5. Plot 20 also has a moderate negative relationship which is also evident.

```
> distplot(dist.vals1)
```

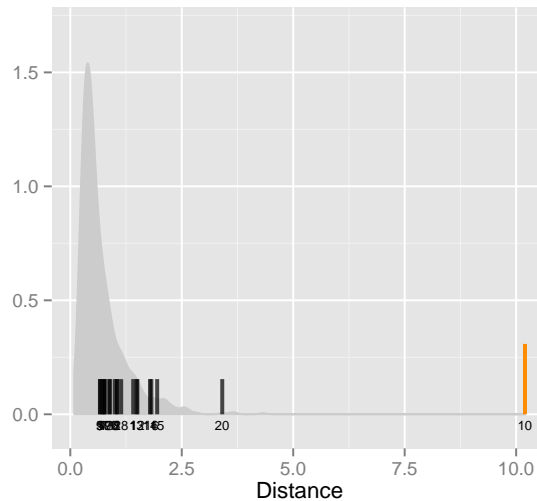


Figure 4.5 Distribution of the **regression based distance** with the distance for the null plots of the lineup represented in black and the distance for the plot of the actual data represented by orange. The distance for the actual plot is much higher than the null plots and the actual plot can be easily identified as seen in Figure 4.3.

The empirical distribution of the binned distance metric is plotted. The number of bins used is 6 on both axes. The bins are selected randomly in this case which had a huge effect on the binned distance. Figure 4.6 shows that the plot of the actual data is unidentifiable which is incorrect.

```
> dist.vals2 <- distmet(lineup.dat, var = c('mpg', 'wt'), 'bin_dist', null_permute('mpg'),
pos = 10, repl = 1000, dist.arg = list(lineup.dat = lineup.dat, X.bin = 6, Y.bin = 6))
> distplot(dist.vals2)
```

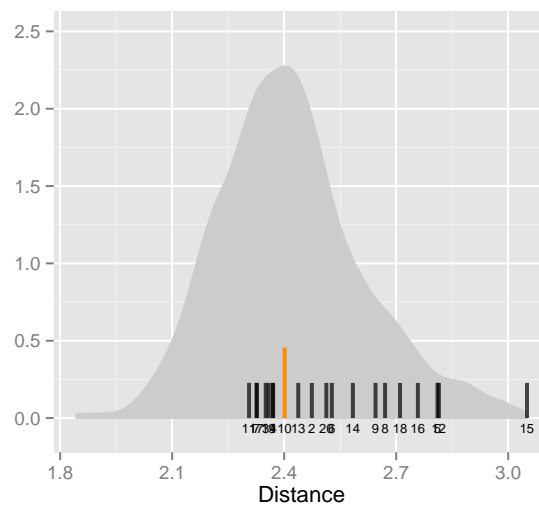


Figure 4.6 Distribution of the **binned distance** with the distance for the null plots of the lineup represented in black and the distance for the plot of the actual data represented by orange. The number of bins used is 6 on both axes. The distance for the actual plot falls within the distances for the null plots. This shows that the lineup is difficult in the sense that the actual cannot be easily identified. Hence the selection of the number of bins is important for binned distance.

The empirical distribution of the binned distance metric is plotted. The number of bins used is 3 on x-axis and 2 on y- axis. The bins are selected using the optimum bin selection procedure. Figure 4.7 shows that the plot of the actual data can easily be identifiable.

```
> dist.vals3 <- distmet(lineup.dat, var = c('mpg', 'wt'), 'bin_dist', null_permute('mpg'),
pos = 10, repl = 1000, dist.arg = list(lineup.dat = lineup.dat, X.bin = 3, Y.bin = 2))
> distplot(dist.vals3)
```

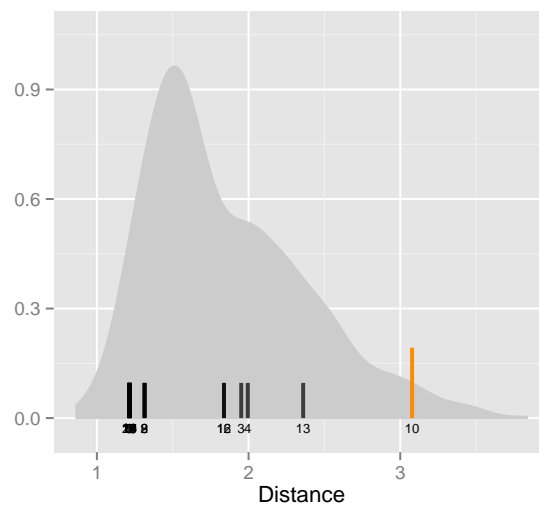


Figure 4.7 Distribution of the **binned distance** with the distance for the null plots of the lineup represented in black and the distance for the plot of the actual data represented by orange. The number of bins used is 3 on x-axis and 2 on y-axis. The number of bins are selected by the optimal bin selection method using Figure 4.4. The distance for the actual plot is larger than the distances for the null plots which matches the lineup.

CHAPTER 5. GENERAL CONCLUSIONS

5.1 General discussion

Visual statistical inference offers promise in situations when there is no formal way of testing hypothesis. In Chapter 2 visual inference procedures were applied to classification techniques in high dimension, low sample size (HDLSS) data. The results suggest that visual inference may be effective for improving the understanding of the emptiness of space in this type of data. With visual inference it was seen that people can visually detect real separation as different from noise up to a reasonably high dimension, for 1D and 2D projections. Visual inference provides a calibration for reading the separation. Although it was not discussed in Chapter 2, we also learned from visual inference that the projection pursuit optimization procedure in `tourr` package is performing correctly. It is possible that visual inference might be used to calibrate results of similar algorithms, where the optimization is used to yield visual products, like multidimensional scaling, PCA, independent component analysis (ICA) and local linear embeddings.

In Chapter 3, we discussed a possible shortcoming of lineup protocol. Unlike classical inference, human subjects have to base their responses on a finite set of null plots although the sampling distribution of null plots is theoretically infinite. Distance metrics were introduced to measure the quality of a lineup and compared to the response of the human subjects. The results suggest that the distance metrics work well in specific situations. The distance metrics considering plot designs works better than generic distance metrics. These metrics may help in designing future Amazon Turk Experiments. Based on the quality of a lineup, more subjects may be allotted to difficult lineups. Comparing different distance metrics on the same lineup may reveal the reason behind the choice made by human subjects. We believe our work with the distance metrics will make lineup

protocol more powerful and will help nullifying the effects of finite null plots.

The R package in Chapter 4 provides open source tools to use visual inference and distance metrics. The package provides ways to generate lineup plots automatically for different null generating mechanisms. It also provides methods to measure the quality of a lineup based on different distance metrics and diagnostic plots to compare several lineups. The users may also use their customized distance metrics to compare lineups. By making this freely available we hope to see increased application of visual inference in situations where there are no conventional testing methods.

5.2 Possibilities for future research

This dissertation provides the ground work for applications of visual inference. There are several natural next steps for this research.

1. Visual inference can be extended to produce confidence intervals. For example, in HDLSS problems to generate a range for the ratio of sample size to dimension at which separation between groups is indistinguishable from noise? In my initial experiment, it was clear that at some point real separation was indistinguishable from noise, and it would be interesting to compare how people determine where this occurs with what has been done analytically on these problems.
2. Visual statistical inference may improve learning in the statistics curriculum. An initial study was done with my undergraduate class and our experimental data suggest that visual inference was very intuitive decision making process which does not require advanced knowledge on mathematics or statistics. It may even be possible to incorporate into elementary school curriculum to build early understanding of randomness.

APPENDIX A. SUPPLEMENTARY MATERIALS OF CHAPTER 2

The material in this document supplement the information presented in Chapter 2. Section A.1 presents the solutions to the lineups used in the chapter. The choice of dimensions used in the Amazon Turk experiment is described in section A.2 .

A.1 Solutions to the Lineups

- The solution to the lineup at Figure 2.2 is Plot 16.
- The solution to the lineup at Figure 2.3 is Plot 8.
- The solution to the lineup at Figure 2.6 is Plot 17.
- The solution to the lineup at Figure 2.7 is Plot 20.

A.2 Choice of dimensions

The experiment is set up with the 3 factors separation, dimension and projection dimension. To decide on the levels of dimension to use, we considered the distribution of the absolute difference of the sample group means, for data with two groups, no separation and projection dimension $d = 1$. The same levels are used for data with 3 groups, $d = 2$, and for data with separation.

Let \mathbf{X}_{ij} denote the j -th observation in the i -th group where $j = 1, \dots, n; i = 1, \dots, g$. The \mathbf{X}_{ij} 's are random noise, generated by drawing samples from a standard normal distribution. For this experiment, $g = 2$ and $n = 15$. The difference between the group means is given by $\bar{X}_1 - \bar{X}_2$ and

$$\bar{X}_1 - \bar{X}_2 \sim \text{Normal}(0, 2/15)$$

Let $U = |\bar{X}_1 - \bar{X}_2|$ where $U \sim \text{Half Normal}$ with scale parameter $\sigma = \sqrt{2/15}$. The expectation and the variance of U are $E(U) = \sigma\sqrt{2/\pi}$ and $Var(U) = \sigma^2(1 - 2/\pi)$, respectively.

For p dimensions, consider p independent samples from the same distribution, denoted as

$$U_m = |\bar{\mathbf{X}}_{m1} - \bar{\mathbf{X}}_{m2}|, \quad m = 1, \dots, p$$

where \mathbf{X}_{mij} is the j -th observation in the i -th group for the m -th dimension. The difference between the two group means projected into one dimension, is the sum over p dimensions of the absolute difference between the means:

$$U = \sum_{m=1}^p U_m = \sum_{m=1}^p |\bar{\mathbf{X}}_{m1} - \bar{\mathbf{X}}_{m2}|$$

and by independence it follows that

$$E(U) = p\sigma\sqrt{2/\pi}, \quad Var(U) = p\sigma^2(1 - 2/\pi)$$

Thus we expect to find this amount of separation between the projected sample means, for data sampled from populations with the same means.

Now consider data where there is some separation (equal to $2c$) between the population means:

$$\mathbf{Z}_{1j} \sim \text{Normal}(-c, 1)$$

$$\mathbf{Z}_{2j} \sim \text{Normal}(c, 1)$$

giving $\bar{Z}_1 - \bar{Z}_2 \sim \text{Normal}(2c, 2/15)$. Then define $Z = |\bar{Z}_1 - \bar{Z}_2|$ where $Z \sim \text{Folded Normal Distribution}$ with scale parameter $\sigma = \sqrt{2/15}$. The expectation and the variance of Z can be calculated to be:

$$E(Z) = \sigma\sqrt{2/\pi} \exp(-2c^2/\sigma^2) + 2c[1 - \Phi(-2c/\sigma)]$$

$$Var(Z) = 4c^2 + \sigma^2 - (E(Z))^2$$

Suppose that only one of the p dimensions is simulated from this distribution, and all of the rest are simulated from populations having identical means. Define V as the sum of the absolute

differences of the mean with one dimension of real separation as

$$V = \sum_{m=1}^{p-1} \mathbf{U}_m + \mathbf{Z}$$

Then, by independence, it follows that:

$$\begin{aligned} E(V) &= (p-1)\sigma\sqrt{2/\pi} + \sigma\sqrt{2/\pi}\exp(-2c^2/\sigma^2) + 2c[1 - \Phi(-2c/\sigma)] \\ \text{Var}(V) &= (p-1)\sigma^2(1 - 2/\pi) + 4c^2 + \sigma^2 - \left(\sigma\sqrt{2/\pi}\exp(-2c^2/\sigma^2) + 2c[1 - \Phi(-2c/\sigma)]\right)^2 \end{aligned}$$

In this experiment, $c = 3$ and $\sigma^2 = 2/15$. Therefore,

$$\exp(-2c^2/\sigma^2) \approx 0 \quad \text{and} \quad \Phi(-2c/\sigma) \approx 0$$

Hence,

$$\begin{aligned} E(V) &= (p-1)\sigma\sqrt{2/\pi} + 6 \\ \text{Var}(V) &= (p-1)\sigma^2(1 - 2/\pi) + \sigma^2 \end{aligned}$$

As dimension p increases for a fixed n , the spread of both U and V increases by a factor of p . The means of U and V also increase with a factor of p but the expected value of the difference between U and V stays constant and is independent of dimension (p).

$$E(V - U) = (p-1)\sigma\sqrt{2/\pi} + 6 - p\sigma\sqrt{2/\pi} = 6 - \sigma\sqrt{2/\pi}$$

Two p -dimensional datasets are generated with 30 observations in each dimension. The datasets are then divided into two groups with 15 observations in each group. For one set, data is obtained from random noise and hence there is no real separation between the two groups. But for the other set, one dimension among these p is adjusted so that the data have some real separation between the groups in that dimension. The absolute difference of the means for each group in each of these p dimensions is considered for both datasets. The absolute difference is considered as we are concerned with projections. These absolute differences between the groups are then summed over all the dimensions to obtain the absolute difference of means for the data. This process is repeated 1000 times. These 1000 sum of absolute differences are then plotted for the different values of p .

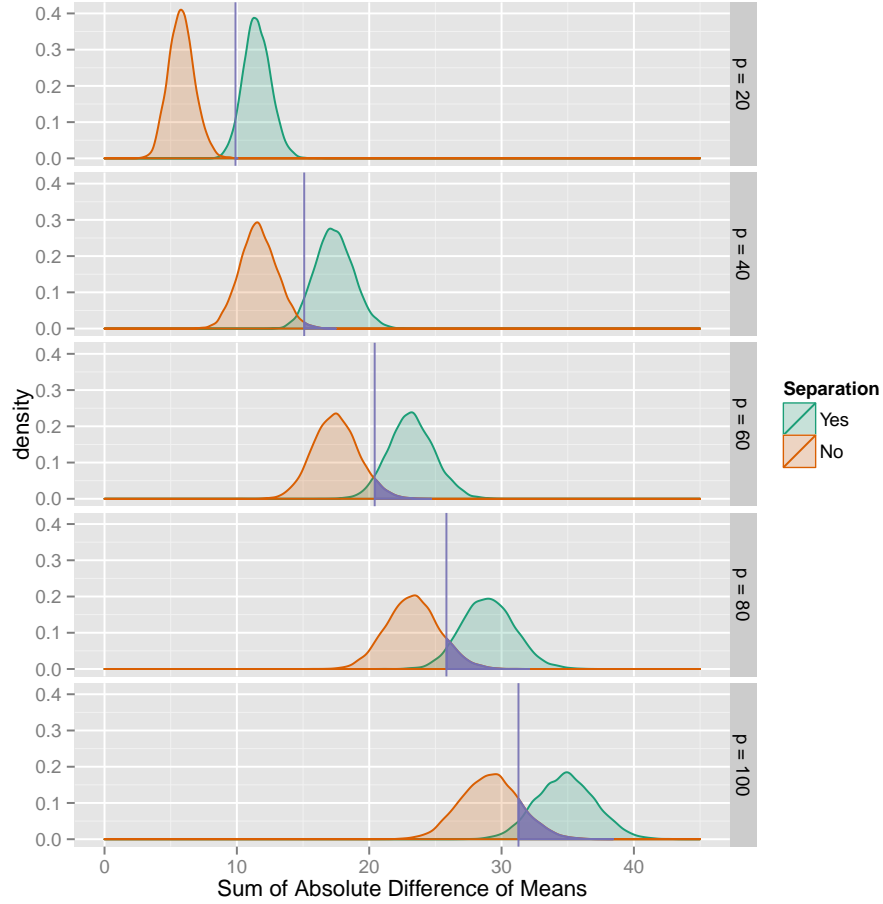


Figure A.1 Plot showing the distribution of the sum of absolute difference of means for data with and without separation for different dimensions. The distributions of data with real separation (V) and purely noise data (U) are shown in brown and green respectively with the dark purple line showing the 5th percentile of V. The dark purple area shows the area of U which is greater than the 5th percentile of V. The dark purple region (δ) increases as dimension (p) increases.

Figure A.1 shows the distribution of sum of absolute difference of means for data with and without separation for different dimensions. The distributions of data with and without separation are shown in brown and green respectively. The area of the distribution of pure noise which is above the 5th percentile of the distribution of data with separation is shown in dark purple. Hence a 5% error is allowed and let the area of the distribution of U greater than the 5th percentile of V be denoted by δ . Mathematically,

$$P[U > V_\alpha] = \delta$$

where V_α is the 100α -th percentile of V, where $\alpha = 0.05$. It can be seen that the dark purple region increases with dimension (p). This indicates that as dimension increases, the distributions of data with or without separation gets closer. Hence it gets harder to detect real differences with higher dimensions. Fixing the area of the dark purple region (δ) and calculating the dimensions to obtain the required region provides the choice of levels of dimension used in the experiment.

The various values of δ are chosen such that the distributions has no separation ($\delta \approx 0$) or has 1%, 5%, 10% and 20% common region. For each value of δ , the procedure is repeated 100 times and Table A.1 shows the summaries of the dimension (p) for each value of δ .

Table A.1 Numerical summaries of dimension p for each value of δ . As the common region δ increases, the median dimension required to obtain the region increases.

δ	Median	5th percentile	95th percentile
0.0000001	24	19	28
0.01	41	38	44
0.02	61	56	64
0.1	77	72	81
0.2	106	99	112

BIBLIOGRAPHY

Amazon (2010). Mechanical Turk.

Anscombe, A. J. (1972). Graphs in Statistical Analysis. *The American Statistician*, 27:1:17–21.

Buja, A., Cook, D., Hofmann, H., Lawrence, M., Lee, E., Swayne, D., and Wickham, H. (2009). Statistical Inference for Exploratory Data Analysis and Model Diagnostics. *Royal Society Philosophical Transactions A*, 367(1906):4361–4383.

Buja, A. and Wolfgang, R. (2005). Calibration for Simultaneity: (Re)Sampling Methods for Simultaneous Inference with Applications to Functional Estimation and Functional Data. Technical report.

Casella, G. and Berger, R. (2002). *Statistical Inference*. Duxbury, California.

Cleveland, W. and McGill, R. (1984). Graphical Perception: Theory, Experimentation and Application to the Development of Graphical Methods. *Journal of the American Statistical Association*, 79(387):531–554.

Comon, P. (1994). Independent Component Analysis: a new concept? *Signal Processing*, 36(3):287 – 314.

Donoho, D. and Jin, J. (2008). Higher Criticism Thresholding: Optimal Feature Selection when Useful Features are Rare and Weak. *Proceedings of the National Academy of Sciences of the United States of America*, 105:14790–14795.

Donoho, D. and Jin, J. (2009). Feature Selection by Higher Criticism Thresholding achieves the Optimal Phase Diagram. *Philosophical Transactions of the Royal Society A*, 367:4449–4470.

- Dudoit, S., Fridlyand, J., and Speed, T. (2002). Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data. *Journal of American Statistical Association*, 97:457:77 – 87.
- Fernholz, L. (2003). Remembering John W. Tukey. *Statistical Science*, 18(3):pp. 336–340.
- Friendly, M. and Denis, D. J. (2001). Milestones in the history of thematic cartography, statistical graphics, and data visualization. (last accessed: April 27, 2011).
- Gelman, A. (2004). Exploratory Data Analysis for Complex Models. *Journal of Computational and Graphical Statistics*, 13(4):755–779.
- Hald, A. (2004). *A History of Parametric Statistical Inference from Bernoulli to Fisher, 1713 to 1935*. ISBN 87-7834-628-2, DK-2100 Copenhagen.
- Hall, P., Marron, J., and Neeman, A. (2005). Geometric Representation of High Dimension, Low Sample Size Data. *Journal of Royal Statistical Society B*, 67:427 – 444.
- Hennig, C. (2010). fpc : Flexible Procedures for Clustering. *R package version 2*.
- Huber, P. J. (1985). Projection Pursuit. *The Annals of Statistics*, 13:435 – 475.
- Huttenlocher, D., Klanderman, G., and Rucklidge, W. J. (1993). Comparing Images Using the Hausdorff Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:9.
- Jung, S., Sen, A., and Marron, J. S. (2012). Boundary Behavior in High Dimension, Low Sample Size asymptotics of Pca. *Journal of Multivariate Analysis*, 109:190–203.
- Kirk, R. E. (1996). Practical significance: A concept whose time has come. *Educational and psychological measurement*, 56.5:746 – 759.
- Kutner, M., Nachtsheim, C., Neter, J., and Li, W. (2005). *Applied Linear Statistical Models*. McGraw-Hill/Irwin, New York.
- Lee, E.-K. and Cook, D. (2010). A Projection Pursuit Index for Large p Small n Data. *Statistics and Computing*, 20:3:381–392.

- Lehmann, E. L. (1997). *Testing Statistical Hypotheses*. Springer, New York.
- Majumder, M., Hofmann, H., and Cook, D. (2013). Validation of Visual Statistical Inference, Applied to Linear Models. *Journal of American Statistical Association*, 108:503:942 – 956.
- Marron, J. S., Todd, M. J., and Ahn, J. (2007). Distance Weighted Discrimination. *Journal of American Statistical Association*, 480:1267–1271.
- Moore, D., McCabe, G., Duckworth, W., and Alwan, L. (2009). *The Practice of Business Statistics*. W. H. Freeman and Company, New York.
- R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Ripley, B. D. (1996). *Pattern Recognition and Neural Networks*. Cambridge University Press.
- Roweis, S. and Saul, L. (2000). Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290:2323 – 2326.
- Roy Chowdhury, N., Cook, D., Hofmann, H., and Majumder, M. (2012). Where’s Waldo: Looking Closely at a Lineup. Technical Report 2, Iowa State University, Department of Statistics.
- Roy Chowdhury, N., Cook, D., Hofmann, H., Majumder, M., Lee, E., and Toth, A. (2014). Visual Statistical Inference for High Dimension, Small Sample Size Data. *Computational Statistics: Accepted*.
- Toth, A., Varala, K., Henshaw, M., Rodriguez-Zas, S., Hudson, M., and Robinson, G. (2010). Brain Transcriptomic Analysis in Paper Wasps Identifies Genes Associated with Behaviour across Social Insect Lineages. *Proceedings of the Royal Society of Biological Sciences - B*, 277:2139 – 2148.
- Toth, A., Varala, K., Newman, T., Miguez, F., Hutchison, S., Willoughby, D., Simons, J., Egholm, M., Hunt, J., Hudson, M., and Robinson, G. (2007). Wasp Gene Expression Supports an Evolutionary Link between Maternal Behavior and Eusociality. *Science*, 318:441 – 444.

- Tukey, J. W. (1977). *Exploratory Data Analysis*. Addison-Wesley Publishing Company, Reading, MA.
- Wickham, H. (2009). *ggplot2: Elegant Graphics for Data Analysis*. Springer New York.
- Wickham, H., Cook, D., Hofmann, H., and Buja, A. (2010). Graphical Inference for Infovis. *IEEE Transactions on Visualization and Computer Graphics*, 16.
- Wickham, H., Cook, D., Hofmann, H., and Buja, A. (2011). tourr: An R Package for Exploring Multivariate Data with Projections. *Journal of Statistical Software*, 40.
- Wilkinson, L. (1999). *The Grammar of Graphics*. NY: Springer, New York.
- Wilkinson, L., Anand, A., and Grossman, R. L. (2005). Graph-theoretic scagnostics. In *INFOVIS*, volume 5, page 21.
- Witten, D. and Tibshirani, R. (2011). Penalized Classification using Fisher’s Linear Discriminant. *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 73(5):753 – 772.
- Yata, K. and Aoshima, M. (2011). Effective PCA for High Dimension, Low Sample Size Data with Noise Reduction via Geometric Representations. *Journal of Multivariate Analysis*, 105:193–215.
- Zhao, Y., Cook, D., Hofmann, H., Majumder, M., and Roy Chowdhury, N. (2012). Mind Reading Using an Eyetracker to See How People Are Looking at Lineups. *The American Statistician: Submitted*.