

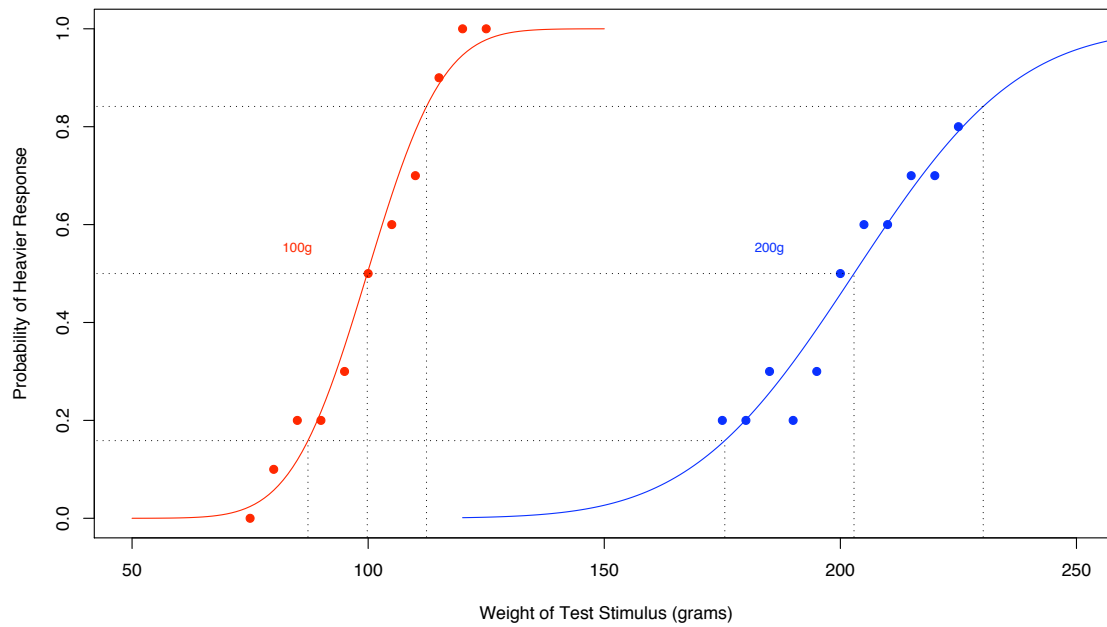
Psychology of Perception

Psychology 4165, Spring 2009

Laboratory 1

Noisy Representations

Figure 1: Weight Discrimination Psychometric Functions



Lab 1: Noisy Representation

Page intentionally blank

Lab 1: Noisy Representation

Introduction

Classical methods of psychophysics involve the measurement of two types of sensory thresholds: the absolute threshold, RL (*Reiz Limen*), the weakest stimulus that is *just* detectable, and the difference threshold, DL (*Differenz Limen*), the smallest stimulus increment away from a standard stimulus that is *just* detectable (also called the Just-Noticeable Difference, the JND). Gustav Theodor Fechner (1801–1887), in *Elemente der Psychophysik* (Fechner, 1860) introduced three psychophysical methods for measuring absolute and difference (JND) thresholds: the method of adjustment; the method of limits; the method of constant stimuli.

The purpose of this laboratory is to give you experience with the measurement and computation of the JND for lifted weights using the method of constant stimuli and to test the predictions of Weber's Law (see below).

Experiment

You will determine difference thresholds for weight discrimination using the method of constant stimuli for two different standard weights: 100 and 200 grams. One of the foundations of psychophysics is Weber's Law. It states that the difference limen is a constant proportion of the standard:

$$\frac{\Delta I}{I} = k \quad \text{Weber's Law}$$

In this experiment you will test the hypothesis that Weber's constant is the same for two different standard weights, thus validating Weber's Law.

Procedure

In the method of constant stimuli, a standard stimulus is compared a number of times with other fixed stimuli of slightly different magnitude. When the difference between the standard and the comparison stimulus is large, the subject nearly always can correctly choose the heavier of the two weights. When the difference is small, errors are often made. The difference threshold is the transition point between differences large enough to be easily detected and those too small to be detected.

Each of you will serve in three different roles in this experiment: as subject, experimenter, and data recorder. You will use the method of constant stimuli to measure your ability to discriminate small differences in weight using two standard weights: 100 grams and 200 grams. Since there are two different standard weight conditions there are two possible testing orders. You should test yourself in the order assigned to you (100 gram first if your birthday falls on an odd numbered data; 200 gram first if even numbered).

Lab 1: Noisy Representation

Testing Orders

Order No	1st	2nd
1	100	200
2	200	100

The experimenter (E) presents a pair of weights (the standard and one of the test weights listed on the data sheet) to the observer or subject (S). E arranges a support for the preferred arm of S so that S's hand will extend over one weight. By flexion of the wrist S should be able to pick up the weight that E has placed in the appropriate position. S should lift the standard weight first (100 or 200 grams) and then the test weight and judge whether the test weight was heavier or lighter than the standard. The person recording the data needs to know the actual weight of the test weight. If the test weight was judged heavier than the standard, a + should be recorded on the data sheet. If the test weight was judged lighter than the standard, a - should be recorded on the data sheet. The 11 test weights should be scrambled on the table and presented in a "random" order. After all 11 are presented the experimenter should shuffle them again and repeat the presentation of each test weight.

Ideally S should be blindfolded, or at least turn his/her face away so that he/she gets no visual cue as to which weight is presented. E then gives S the necessary instructions:

1. "When I say, 'Now,' lift the standard weight which is directly below your hand, using a wrist motion. Notice its weight, return it to the table, and lift your hand again. When I repeat, 'Now,' do the same with the second weight."
2. "Report whether or not the test (second) weight seems **heavier** than the standard weight. **Do not give 'equal' judgments.** Guess if you are not certain."
3. E should present the weights in pairs, placing first the standard weight (either 100 or 200 grams, as appropriate) and then a test weight directly below S's hand. E should try to develop a regular rhythm. Stimuli should be presented for 1–2 sec. duration, separated by an equal period of time. The intervals between pairs should be longer. Allow the subject to rest from time to time. At least 10 presentations of each test stimulus should be made. The order of the test weights should be randomized.

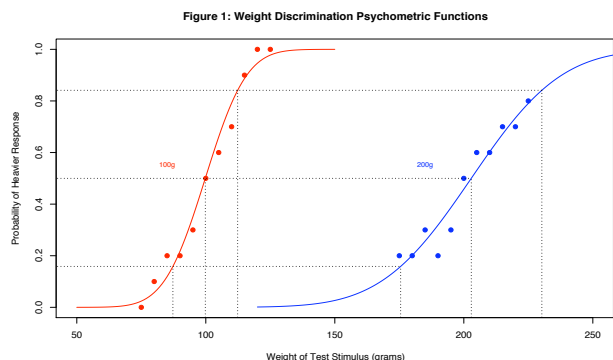
Data Tabulation and Analysis

1. Transfer the frequency of "heavier" and "lighter" judgments for each test weight from your data sheet to the summary sheet at the end of this handout. Compute the total of heavier and lighter judgments and then compute the

Lab 1: Noisy Representation

probability of making a heavier judgment.

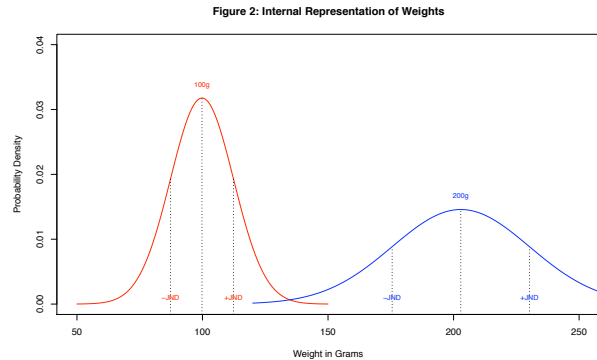
2. Now prepare a data file containing your data. Open the template data file “lab1_glm.txt” by double-clicking on it. It should open in Microsoft Excel. Replace the last two data columns with your own data (frequency of heavier and lighter judgments). Replace the “Lew” in column 1 with your own name (no spaces). When you are finished, choose SaveAs... from the File menu and save the file as a *Text File* with your own name (e.g., harvey_glm.txt). Do not use spaces in the file name.
3. Use the R commands listed in the file “lab1_glm.R” (reproduced at the end of this handout) to carry out your data analysis (R Development Core Team, 2008). There are two basic steps to the analysis: 1) the generalized linear model function of R (`glm()`) is used to fit a smooth, s-shaped psychometric function to your 100 gram and 200 gram data; and 2) graphs of your results are plotted in two separate figures.
4. The results of the curve-fitting are stored in R objects `glm100` and `glm200`. These results may be viewed using the `summary()` command: `summary(glm100)` and `summary(glm200)`. You can copy the badness-of-fit index, AIC, from your screen to your data sheet. The mean and standard deviations of the best-fitting Gaussian distributions are in R objects `mu100`, `sd100`, `mu200`, and `sd200` respectively.
5. **The JND:** There are two ways to estimate the JND. One way is to compute the reciprocal of the steepness of the best-fitting psychometric function. The steepness is given by the `glm` coefficient corresponding to `Test_Weight`. So the steeper the function, the smaller the JND. Computed this way, one JND is equivalent to one standard deviation of the Gaussian distribution underlying the psychometric function. The second, equivalent method, is to use the difference, in grams, between the weight corresponding to the 0.84 point on the ordinate, and the weight corresponding to the 0.16 point on the ordinate divided by 2.0. Compute the Weber fraction for the 100 and the 200 gram standard by dividing the appropriate JND by the corresponding standard weight. Is the Weber fraction constant?
6. Prepare two graphs illustrating your results. Figure 1 should be a plot of your observed psychometric function data for the 100 and 200 gram standards along with the best-fitting s-shaped psychometric function. The graphic commands to make the figure on the front of this handout are given in the file



Lab 1: Noisy Representation

“lab1_glm.R”. Feel free to use `help(plot)` and modify the plotting parameters to achieve the kind of plot that appeals to you. Your Figure 1 should look like the graph above. You save a graph in R as a file by clicking on the graph window and choosing save from the File menu.

7. The second figure plots the noisy representations of the 100 and 200 gram standard weights using `dnorm()`. Check out the graphing commands in the `lab1_glm.R` file. Your Figure 2 should like something like this:



8. **Hypothesis Testing:** We will assemble your individual data into a single data file that will be available for the next lab meeting. Test the hypothesis, using R, that the value of the Weber fraction is the same for 100 grams as for 200 grams. The appropriate analysis is a repeated measures analysis of variance.
9. When you have a lot of data from different people it is a good idea to make graphs of them so you get an idea what they look like and whether or not there are differences among different groups or levels of factors. This strategy is part of what is called exploratory data analysis (Tukey, 1977). Three such plots are called histograms, strip charts and box plots. The R commands to produce them are given in the file “lab1_aov.R” and are reproduced at the end of this handout.

Lab Report

Your lab report should be brief and contain five sections: cover sheet, introduction, methods, results, and discussion. These sections should conform to the American Psychological Association (APA) style (American Psychological Association, 2001) as described in Chapter 13 of the Martin textbook (Martin, 2007). The results section should have the graphs described above and a table giving the JND and Weber’s k for the 100 and 200 gram conditions. Do your results support Weber’s Law?

The report is due at the beginning of lab meeting (**3 & 5 February 2009**). Late labs will receive a grade of zero. All lab reports must be prepared with a word processor. This lab report is worth 30 points.

Lab 1: Noisy Representation

References

- American Psychological Association (2001). *Publication Manual of the American Psychological Association* (5th ed.). Washington, DC: American Psychological Association.
- Fechner, G. T. (1860). *Elemente der Psychophysik*. Leipzig, Germany: Breitkopf and Härtel.
- Martin, D. W. (2007). *Doing psychology experiments* (7th ed.). Belmont, CA: Thomson Wadsworth.
- R Development Core Team (2008). *R: A language and environment for statistical computing*. from <http://www.R-project.org/>.
- Tukey, J. W. (1977). *Exploratory data analysis*. Reading, MA: Addison-Wesley.

Lab 1: Noisy Representation

Method of Constant Stimuli Data Sheet — 100 gram Standard											
Trial No.	75	80	85	90	95	100	105	110	115	120	125
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
Frequency of Test "Heavier"											
Frequency of Test "Lighter"											

Note: The judgment of the observer should be which weight (Test or Standard) is heavier.

Lab 1: Noisy Representation

Method of Constant Stimuli Data Sheet — 200 gram Standard											
Trial No.	175	180	185	190	195	200	205	210	215	220	225
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
Frequency of Test "Heavier"											
Frequency of Test "Lighter"											

Note: The judgment of the observer should be which weight (Test or Standard) is heavier.

Lab 1: Noisy Representation

Data Summary Sheet

Weight in grams	No. Heavier than Standard	No. Lighter than Standard	Total No. of Judgments
75			
80			
85			
90			
95			
100			
105			
110			
115			
120			
125			
175			
180			
185			
190			
195			
200			
205			
210			
215			
220			
225			

Lab 1: Noisy Representation

Curve-Fitting Summary from glm()

		Mean		Std Deviation		Index-of-Fit: AIC	
Name	Order	mu100	mu200	sd100	sd200	100 g	200 g

Please transfer your results above to the master data sheet for the class and we will prepare a master file with everyone's data for analysis next meeting. Your name will not appear on the master file.

Lab 1: Noisy Representation

```
# PSYC 4165 Lab 1
# Generalized Linear Model (glm)
# command file for using the R command glm() to compute the best-
# fitting
# Gaussian integral (Gaussian CDF) to psychometric function data.
# Lewis O. Harvey, Jr.
# Department of Psychology
# University of Colorado

par(ask=TRUE) # ask before plotting the next graph

# *****
# Analyze the data
# *****

fn <- file.choose()
pf <- read.delim(fn, header=TRUE)

# compute the probability of making a "heavier" judgment for each
# of the test stimuli
pf <- cbind(pf, p_Heavier = pf$No_Heavier / (pf$No_Heavier +
pf$No_Lighter))

# put the two sets of data (100 and 200 gram) into separate data frames
pf100 <- subset(pf, Standard=="100_gr")
pf200 <- subset(pf, Standard=="200_gr")

# compute the best-fitting Gaussian integrals for each set of data
# using a generalized linear model (glm).
# The "probit" link specifies that a Gaussian probability distribution
# is being fit to the data.
# The "binomial" family specifies that the dependent variable
# (number of heavier and lighter judgments) are drawn from
# a binomial distribution.
glm100 <- glm(cbind(No_Heavier, No_Lighter) ~ Test_Weight,
  data = pf100,
  family = binomial(link = "probit"))
glm200 <- glm(cbind(No_Heavier, No_Lighter) ~ Test_Weight,
  data = pf200,
  family = binomial(link = "probit"))

# compute the mean and standard deviations of the Gaussian
# probability density functions from the glm solutions

# mean of the 100 gram representation
mu100 <- -coefficients(glm100)[1]/coefficients(glm100)[2]
# standard deviation of the 100 gram representation
sd100 <- 1 / coefficients(glm100)[2]
# mean of the 200 gram representation
mu200 <- -coefficients(glm200)[1]/coefficients(glm200)[2]
# standard deviation of the 200 gram representation
sd200 <- 1 / coefficients(glm200)[2]

# get the cumulative probability corresponding to plus and minus 1
```

Lab 1: Noisy Representation

```
standard deviation
# that we define to correspond to 1 JND (these should be 0.16 and 0.84)
pmJND <- pnorm(-1.0) # probability 1 sd below the mean
ppJND <- pnorm( 1.0) # probability 1 sd above the mean

# weights corresponding to 1 JND above and below the means
q1 <- qnorm(pmJND, mu100, sd100) # lower jnd weight for 100 grams
q2 <- qnorm(ppJND, mu100, sd100) # upper jnd weight for 100 grams
q3 <- qnorm(pmJND, mu200, sd200) # lower jnd weight for 200 grams
q4 <- qnorm(ppJND, mu200, sd200) # upper jnd weight for 200 grams

# compute the JNDs from lower and upper 1 sd points
jnd100 <- (q2 - q1) / 2 # jnd in grams
jnd200 <- (q4 - q3) / 2 # jnd in grams

# compute the Weber fraction for 100 and 200 grams
web100 <- jnd100 / 100
web200 <- jnd200 / 200

# *****
# Plotting Section
# *****

# test stimuli used under the 100 and 200 standard conditions
ts100 <- c(75, 80, 85, 90, 95, 100, 105, 110, 115, 120, 125)
ts200 <- c(175, 180, 185, 190, 195, 200, 205, 210, 215, 220, 225)

# plot Figure 1, the psychometric functions
plot(p_Heavier ~ Test_Weight, data = pf100, col = "red",
     xlim = c(50, 250), ylim = c(0, 1), pch = 19,
     xlab="Weight of Test Stimulus (grams)",
     ylab="Probability of Heavier Response",
     main="Figure 1: Weight Discrimination Psychometric Functions")

points(p_Heavier ~ Test_Weight, data = pf200, pch = 19, col = "blue")

# plot the predicted smooth curves
x100 <- 50:150
y <- predict(glm100, data.frame(Test_Weight=x100), type = "response")
lines(y ~ x100, col = "red")
x200 <- 120:280
y <- predict(glm200, data.frame(Test_Weight=x200), type = "response")
lines(y ~ x200, col = "blue")

# label each line
text( 85, .55, "100g", col="red", cex=0.75)
text(185, .55, "200g", col="blue", cex=0.75)

# plot the alpha markers (the point of subjective equality)
lines(c(mu100, mu100), c(-0.1, 0.5), lty=3)
lines(c(mu200, mu200), c(-0.1, 0.5), lty=3)
lines(c(40, mu200), c(0.5, 0.5), lty=3)

# now draw the vertical JND lines (lty=3 makes the line dashed)
```

Lab 1: Noisy Representation

```
lines(c(q1, q1), c(-0.1, pmJND), lty=3)
lines(c(q2, q2), c(-0.1, ppJND), lty=3)
lines(c(q3, q3), c(-0.1, pmJND), lty=3)
lines(c(q4, q4), c(-0.1, ppJND), lty=3)

# draw the vertical JND probability lines
lines(c(40, q3), c(pmJND, pmJND), lty=3)
lines(c(40, q4), c(ppJND, ppJND), lty=3)

# plot Figure 2, Gaussian stimulus representations
plot(x100, dnorm(x100, mu100, sd100),
     type="l", col = "red",
     xlim = c(50, 250), ylim = c(0, 0.04),
     xlab = "Weight in Grams",
     ylab = "Probability Density",
     main = "Figure 2: Internal Representation of Weights")
lines(x200, dnorm(x200, mu200, sd200), col = "blue")

# probability density at each mean
dmu100 <- dnorm(mu100, mu100, sd100)
dmu200 <- dnorm(mu200, mu200, sd200)
lines(c(mu100, mu100), c(-0.1, dmu100), lty=3)
lines(c(mu200, mu200), c(-0.1, dmu200), lty=3)

# probability densities at each JND point
d1 <- dnorm(q1, mu100, sd100)
d2 <- dnorm(q2, mu100, sd100)
d3 <- dnorm(q3, mu200, sd200)
d4 <- dnorm(q4, mu200, sd200)

# now draw the vertical JND lines (lty=3 makes the line dashed)
lines(c(q1, q1), c(-0.1, d1), lty=3)
lines(c(q2, q2), c(-0.1, d2), lty=3)
lines(c(q3, q3), c(-0.1, d3), lty=3)
lines(c(q4, q4), c(-0.1, d4), lty=3)

# label each distribution
text(mu100, dmu100+0.002, "100g", col="red", cex=0.75)
text(mu200, dmu200+0.002, "200g", col="blue", cex=0.75)

# label the JND markers
text(q1, 0.001, "-JND", col="red", cex=0.75)
text(q2, 0.001, "+JND", col="red", cex=0.75)
text(q3, 0.001, "-JND", col="blue", cex=0.75)
text(q4, 0.001, "+JND", col="blue", cex=0.75)

par(ask=FALSE)
```

Lab 1: Noisy Representation

```
# PSYCH 4165 Lab 1
# command file for computing a repeated measures
# analysis of variance using lme() in R.
# Lewis O. Harvey, Jr.
# Department of Psychology
# University of Colorado
# 26 January 2009

# load the nlme library to use lme()
library(nlme)

# set graphic parameters to plot three graphs in one panel
par(mfcol = c(3,1))
# ask before plotting
par(ask=TRUE)

# read in group data in wide format (from Excel text file)
fn <- file("lab_1_group_data_wide.txt")
df.wide <- read.delim(fn, header=TRUE)

# *****
# reshape the file into long format
# Note: reshape() is a bit of a pain to use but what it does
# is take data that are in a so-called wide format
# (one row per subject, separate columns per dependent variable)
# and put it into the long formate needed by many R analysis routines
# (multiple rows per subject, single column per dependent variable)
df.long <- reshape(df.wide,
  varying = list(c("mu100", "mu200"), c("sd100", "sd200"), c("aic100",
"aic200")),
  idvar = "subject",
  times = c("100", "200"),
  v.names = c("mu", "sd", "aic"),
  timevar = "standard",
  direction = "long")
rownames(df.long) <- 1:dim(df.long)[1]
df.long$standard <- factor(df.long$standard) # make standard a factor
# *****

df <- df.long # use shorter name for the data frame

# *****
# Calculate Weber's Constant (k) for each subject and
# compute the log of Weber's constant
# Add both to the data frame
df <- cbind(df, Weber_k = df$sd / as.numeric(as.character((df$standard))))
df <- cbind(df, Log_k = log10(df$Weber_k))

# Write out a summary of the variables in the data frame:
summary(df)

# *****
# plotting section
# *****
# Make a histogram of the Weber k values
with(df, hist(Weber_k, breaks=10,
  xlab = "Weber's k",
  ylab = "Frequency",
  main = "Histogram of Lab 1 Weber's k"))

# Make a strip chart of Weber's k for each order
```

Lab 1: Noisy Representation

```
with(df, stripchart(Weber_k ~ order,  
  method = "jitter", jitter = 0.03,  
  xlab = "Weber's k",  
  ylab = "Testing Order",  
  main = "Strip Chart of Lab 1 Weber's k"))  
  
# Make a box plot of Weber's k for each order  
boxplot(Weber_k ~ order, data = df,  
  horizontal = TRUE,  
  ylab = "Testing Order",  
  xlab = "Weber's k",  
  main = "Box and Whisker Plot of Lab 1 Weber's k")  
  
# *****  
  
# Make a histogram of the data  
with(df, hist(Weber_k, breaks=10,  
  xlab = "Weber's k",  
  ylab = "Frequency",  
  main = "Histogram of Lab 1 Weber's k"))  
  
# Make a strip chart of Weber's k for each standard  
with(df, stripchart(Weber_k ~ standard,  
  method = "jitter", jitter = 0.03,  
  xlab = "Weber's k",  
  ylab = "Standard Weight",  
  main = "Strip Chart of Lab 1 Weber's k"))  
  
# Make a box plot of Weber's k for each standard  
boxplot(Weber_k ~ standard, data = df,  
  horizontal = TRUE,  
  ylab = "Standard Weight",  
  xlab = "Weber's k",  
  main = "Box and Whisker Plot of Lab 1 Weber's k")  
# *****  
  
# compute the means  
print(" ", quote = FALSE)  
print("means", quote = FALSE)  
print(with(df, tapply(Weber_k, order, "mean")))  
print(with(df, tapply(Weber_k, standard, "mean")))  
  
# use lme() to compute the ANOVA  
print(" ", quote = FALSE)  
print("Results of lme() Analysis Of Variance", quote = FALSE)  
  
a <- lme(Weber_k ~ standard * order, data = df, random = ~ 1 | subject)  
print(anova(a))  
  
# return plotting to one graph per panel  
par(mfcol = c(1,1))  
par(ask=FALSE)
```