StatR 101: Fall 2012

Homework 1

Rod Doe

1. Write a few sentences outlining your background in programming and in statistics.

I have been a professional software developer for 22 years, with the majority of that time spent working on machine vision systems. Initially, my work was mostly in C/C++, but I have transitioned over to doing mostly database work. I just finished a tour of duty at doing Business Intelligence at Microsoft, and I am currently employed by Apex Learning in Seattle as a Senior SQL engineer.

While I took a statistics class in the mechanical engineering curriculum at UW, I would describe myself as more recently self-taught in statistics, trying to keep my head above water in the Six Sigma environments in the semiconductor industry. I first used R to correlate the image acquisition behavior of two different machine vision systems as they quantified the position and size of test probe marks on silicon wafer test pads.

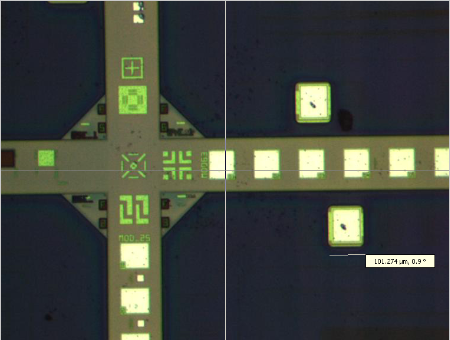


Figure 1: Test Probe marks on silicon wafer test pads

In this application, test probes (size on the order of 10 microns) contact a test pad (size on the order of 100 microns, the approximate diameter of a human hair) simultaneously on hundreds of sites on a single integrated circuit (IC), on hundreds of ICs on a wafer. I used R to correlate the results of a low-resolution, high-speed defect detection system with those of a dedicated high-resolution probe mark analysis system. It turned out that the machines correlated reasonably well – a Pareto chart of results generally had the same order. The machines did not correlate well on pads with multiple probe marks. In that case, one machine chose the biggest mark as the probe mark, and the other machine drew a bounding box around the probe marks and called that the probe mark. Sigh…

1. Describe some kinds of data that are of interest to you, either professionally, personally or (if nothing comes to mind) something that is publicly available. Describe briefly the variables of interest (i.e. categorical or quantitative), what the ranges and possible values of the data area, and what relationships there may or may not be between them.

I am interested in somehow quantifying the effect of a particular member of the Seattle Sounders, Mauro Rosales. I observe that the team performs better when Rosales is on the pitch, but he does not score many goals or contribute significantly to the defense. I want to somehow understand/quantify the Rosales effect. Data available include:

* The minutes in which Rosales (and other players) entered and left the match.
* Positions of players. (This may not be specifically available.)
* Goals scored by the Sounders.
  + By player
  + By minute
* Goals scored by the opposition.
  + By player
  + By minute

When I grow up, I want to be a statistician for the Sounders. I have a hunch, though, that you can’t do both.

1. Create a vector of all the even numbers from 2 to 100 using the constraints given below:

using the seq() function:

x <-seq(from=2, to=100, by=2)

using the `:' operator:

x <- 2:100

using the cumsum() function.

x <- cumsum(rep(2, 50))

1. Copy and paste the following code into R, which will generate a list of names, sex, and grades for students in a class.

Names <- c("Alana", "Bettie", "Consuela", "Dona", "Elaine", "Frances", "Gerri", "Helene", "Ichabod", "Jin", "Kenyatta", "Larry", "Mikhailo","Nick", "Odin")

Sex <- c(rep("F",8), rep("M",7))

Grades <- round(runif(15,50,100))

1. Use the names() function to assign the students names to the grades and output the result.

names(Grades) <- Names

names(Grades)

Grades

> names(Grades) <- Names

> names(Grades)

[1] "Alana" "Bettie" "Consuela" "Dona" "Elaine" "Frances" "Gerri" "Helene"

[9] "Ichabod" "Jin" "Kenyatta" "Larry" "Mikhailo" "Nick" "Odin"

> Grades

Alana Bettie Consuela Dona Elaine Frances Gerri Helene Ichabod Jin

60 98 71 90 61 89 58 73 91 92

Kenyatta Larry Mikhailo Nick Odin

57 82 86 96 59

1. Extract the names of the students (if any) with grades greater than or equal to 90%.

> which(Grades >= 90)

Bettie Dona Ichabod Jin Nick

2 4 9 10 14

> names(which(Grades >= 90))

[1] "Bettie" "Dona" "Ichabod" "Jin" "Nick"

1. Extract the names of the students (if any) with grades under 60%.

> names(which(Grades < 60))

[1] "Gerri" "Kenyatta" "Odin"

1. Extract the grades of male students and the female students and create new objects called Grades.M and Grades.F.

> Grades[which(Sex == "F")]

Alana Bettie Consuela Dona Elaine Frances Gerri Helene

60 98 71 90 61 89 58 73

> Grades.F = Grades[which(Sex == "F")]

> Grades.F

Alana Bettie Consuela Dona Elaine Frances Gerri Helene

60 98 71 90 61 89 58 73

> Grades.M = Grades[which(Sex == "M")]

> Grades.M

Ichabod Jin Kenyatta Larry Mikhailo Nick Odin

91 92 57 82 86 96 59

1. Use the sum() and length() functions to obtain average grades of the male and female students. Which have higher grades?

> sum(Grades.M)

[1] 563

> length(Grades.M)

[1] 7

> avg.M = sum(Grades.M)/length(Grades.M)

> avg.M

[1] 80.42857

> avg.F = sum(Grades.F)/length(Grades.F)

> avg.F

[1] 75

Males have higher average grades.

1. More grade manipulation: Suppose that I am teaching a graduate statistics course with only 5 students. Below is a list of their scores in this class.

Exam Alice Bruno Carl Dolores Ebenezer

quiz 1 89 95 75 82 90

quiz 2 87 90 60 0 90

midterm 84 92 72 88 96

final 78 76 58 68 80

1. Create a vector for each of the tests. The vectors should contain the 5 student scores for that test (i.e. quiz1,quiz2, midterm, final) as shown in Table 1.

> Names <- c("Alice", "Bruno", "Carl", "Dolores", "Ebenezer")

> quiz.1 <- c(89, 95, 75, 82, 90)

> quiz.2 <- c(87, 90, 60, 0, 90)

> midterm <- c(84, 92, 72, 88, 96)

> final <- c(78, 76, 58, 68, 80)

1. Use the cbind() function to combine the vectors to make a matrix with the test scores reported in rows, individual student's scores in columns. Call this matrix ClassScores.

(Assumed cbind meant rbind)

> ClassScores <- rbind(quiz.1, quiz.2, midterm, final)

> ClassScores

[,1] [,2] [,3] [,4] [,5]

quiz.1 89 95 75 82 90

quiz.2 87 90 60 0 90

midterm 84 92 72 88 96

final 78 76 58 68 80

1. Name the columns of this matrix (again, using names()) to reflect the column names in Table 1.

This did not have the desired effect:

> names(ClassScores) <- Names

> ClassScores

[,1] [,2] [,3] [,4] [,5]

quiz.1 89 95 75 82 90

quiz.2 87 90 60 0 90

midterm 84 92 72 88 96

final 78 76 58 68 80

attr(,"names")

[1] "Alice" "Bruno" "Carl" "Dolores" "Ebenezer" NA

[7] NA NA NA NA NA NA

[13] NA NA NA NA NA NA

[19] NA NA

However, this worked:

> names(quiz.1) <- Names

> quiz.1

Alice Bruno Carl Dolores Ebenezer

89 95 75 82 90

> ClassScores <- rbind(quiz.1, quiz.2, midterm, final)

> ClassScores

Alice Bruno Carl Dolores Ebenezer

quiz.1 89 95 75 82 90

quiz.2 87 90 60 0 90

midterm 84 92 72 88 96

final 78 76 58 68 80

1. Suppose Alice wants a list of her test scores in the class. Create a vector using the matrix ClassScores with the test scores for only Alice, call it AliceProgress.

> AliceProgress <- ClassScores[,"Alice"]

> AliceProgress

quiz.1 quiz.2 midterm final

89 87 84 78

1. Create a new vector using the matrix ClassScores that contains the average quiz scores for each student (average for quiz 1 and quiz 2), call it QuizAverage.

> QuizScores <- ClassScores[1:2,]

> QuizScores

Alice Bruno Carl Dolores Ebenezer

quiz.1 89 95 75 82 90

quiz.2 87 90 60 0 90

> ClassScores[1,]

Alice Bruno Carl Dolores Ebenezer

89 95 75 82 90

> ClassScores[1:2,]

Alice Bruno Carl Dolores Ebenezer

quiz.1 89 95 75 82 90

quiz.2 87 90 60 0 90

> QuizAverage <- (ClassScores[1,] + ClassScores[2,]) / 2

> QuizAverage

Alice Bruno Carl Dolores Ebenezer

88.0 92.5 67.5 41.0 90.0

1. Use the rbind() function to create a new matrix with the QuizAverage vector attached to ClassScores as the last row, call it ClassScores2.

> ClassScores2 <- rbind(quiz.1, quiz.2, midterm, final, QuizAverage)

> ClassScores2

Alice Bruno Carl Dolores Ebenezer

quiz.1 89 95.0 75.0 82 90

quiz.2 87 90.0 60.0 0 90

midterm 84 92.0 72.0 88 96

final 78 76.0 58.0 68 80

QuizAverage 88 92.5 67.5 41 90

1. Final grades in this class are calculated as follows: quizzes 20%, midterm 25%, final 55%. Use the matrix ClassScores2 to create a vector that contains the students overall scores in the class, call it FinalGrades.

> FinalGrades <- (0.2 \* ClassScores2[5,] + 0.25 \* ClassScores2[3,] + 0.55 \* ClassScores2[4])

> FinalGrades

Alice Bruno Carl Dolores Ebenezer

81.5 84.4 74.4 73.1 84.9

1. Use the mean() and sd() function in R to calculate the mean and standard deviation for the final grades.

> mean(FinalGrades)

[1] 79.66

> sd(FinalGrades)

[1] 5.568034

Bonus problem: Read about the estimation of standard deviations on Wikipedia: http://en.wikipedia.org/wiki/Standard\_deviation#Estimation. The article confusingly

defines two estimates: the standard deviation of the sample, and the sample standard deviation.

1. What formula does R use when calculating standard deviation (type out the formula)? Is this the appropriate formula to for calculating standard deviation in this example?

The standard deviation is the square root of the variance. The variance of a set of data x of N data points is defined as:

Variance(x) = ∑ (y – mean(x))2 /D,

Where D = N – 1 for a sample, and D = N for a population.

The var() function in R calculates the variance of a vector. The question is which value of D does it ues? A simple experiment reveals the answer:

> x = c(5,3,5,3)

> mean(x)

[1] 4

> var(x)

[1] 1.333333

> sd(x)

[1] 1.154701

The sum of the squares of the vector X is 4. R calculates the variance as 4/3, so R uses the sample version of the variance in the var() function.

(b) Compute the sd for the class final grades without using the sd() function. What formula did you choose to use? How does it compare in magnitude to the formula R uses?

> var(FinalGrades)

[1] 31.003

Take the square root of that to get 5.57.