ECS 132 Final Project

Rajiv Battula
Alex Devlin
Edward Huang
Gowtham Vijayaragavan

December 5, 2011
1 Introduction

Using the knowledge we gained in ECS 132, we set out to examine two datasets in hopes of drawing meaningful conclusions. The first set of data was a survey of recent college graduates, administered by the National Center for Science and Engineering Statistics (NCSES), which is a part of the National Science Foundation. This survey gave us an insight into what college graduates look for when seeking job opportunities and how they feel about the positions they eventually accept. More specifically, we targeted Computer Science, Economics and Communications Majors. We looked at what graduates from these majors thought about salaries and found some interesting facts about these three groups of people.

Next, we looked at data that the researchers from the “Stanford Network Analysis Project” (SNAP) gathered from the production recommendations given to users on Amazon.com. Specifically, the researchers looked at the “Customers Who Bought This Item Also Bought” feature on Amazon.com. Researchers from SNAP looked at a product, i, frequently co-purchased with another product, j. Their resultant data consists of edges from a product i to a product j, if product i is often purchased alongside product j.

2 National Survey of Recent College Graduates

This survey was given out to college graduates who received their bachelors or masters degrees between July 1, 2000 and June 30, 2002. Since this survey was administered in 2003, it is outdated, and many significant events have transpired since. This data may not be completely accurate, or representative, of the present day.

2.1 Importance of Compensation

2.1.1 Salary Expectations

We wanted to see how much Computer Science, Economics, and Communication majors value compensation. Firstly, we looked at how much importance graduates of each major place on salaries. With 1 being “Very Important” and 4 being “Not Important At All”, we found that these majors give almost equal importance to salaries. Computer Science Majors gave an average importance of 1.43 to salaries while their Economics counterparts gave an average of 1.48 importance. Lastly, we looked at how much importance Communications majors gave to salary, with an average importance of 1.42.
The 95% confidence intervals for the difference between the Computer Science averages and the other two averages for salary importance are as follows.

<table>
<thead>
<tr>
<th>Difference Between</th>
<th>From</th>
<th>To</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Science - Economics</td>
<td>-0.09185297</td>
<td>-0.01962764</td>
</tr>
<tr>
<td>Computer Science - Communications</td>
<td>-0.03736083</td>
<td>0.05199687</td>
</tr>
</tbody>
</table>

As we can see, the ranges of the confidence intervals are small. This means that our sample of college graduates is fairly representative of the overall population. As for the actual comparison, computer science graduates appear to value their salaries to about the same degree as Communications majors and Economics majors. The difference between Computer Science and Economics may only include negative values, but they are so minuscule that for all intents and purposes, we cannot conclude any difference in how much they value their salaries. The graphs of the densities of each major are in the appendix.

2.1.2 Salary Satisfaction

Next, logically we wanted to find out how satisfied graduates were with their salaries on their current job. Here, we found a bigger gap. Again, with 1 as “Very Satisfied” and 4 as “Very Dissatisfied”, CS majors were more satisfied with their salaries with an average of 1.73. The Economics and Communications majors were less satisfied with an average of 1.90 and 1.98 respectively.
Again, we have the confidence intervals, with 95% confidence, for the differences in the satisfaction of compensation for Computer Scientists with the other two majors.

<table>
<thead>
<tr>
<th>Difference Between</th>
<th>From</th>
<th>To</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Science - Economics</td>
<td>-0.1719048</td>
<td>-0.07435523</td>
</tr>
<tr>
<td>Computer Science - Communications</td>
<td>-0.2631668</td>
<td>-0.143071</td>
</tr>
</tbody>
</table>

The range here is slightly more pronounced, though still not extremely large. Though there is a statistically significant chance that Computer Science graduates are more satisfied with their salaries than Economics and, to a greater extent, Communications graduates, the difference itself in the population may not be great.

We also see that these confidence intervals are similar to the last ones about salary expectations. The confidence intervals are small which means that the sample size selected by the NCSES was quite representative of larger sample size.

### 2.2 Professional Conferences

We wanted to find out how many graduates of each major have been to a professional meeting in the past year (relative to 2003). We looked at the question

“During the past year, did you attend any professional society or association meetings or professional conferences?”
As shown, it seems that a much smaller percentage of Computer Science graduates attend conferences compared to their peers from the Economics and Communications majors.

The confidence interval, with 95% confidence, for this question came out to be as follows.

<table>
<thead>
<tr>
<th>Difference Between</th>
<th>From</th>
<th>To</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Science - Economics</td>
<td>-0.1580744</td>
<td>-0.1573609</td>
</tr>
<tr>
<td>Computer Science - Communications</td>
<td>-0.1677969</td>
<td>-0.1669039</td>
</tr>
</tbody>
</table>

The range of the confidence interval for computer science graduates is much lower than the other two ranges, so that there is little doubt that computer scientists as a whole attend conferences less than the other majors. To find the reasons behind this finding, we would need to observe many other factors about Computer Science Majors.

### 2.3 Job Satisfaction

In this section, we calculate the average of the computer science graduates’ satisfaction in 9 fields. Each of the fields could be ranked from 1, as “Very Satisfied”, to 4, as “Very Dissatisfied.” Thus, the average could also fall between 1 and 4. For the computer science graduates, the mean of these
9 fields averaged together was 1.616, indicated that the majority of computer scientists were fairly satisfied with their jobs. In the survey, there was also a field from 1 to 4 for overall satisfaction with one’s job. We compared this number to the mean of only the “Overall Satisfaction” field got 1.747.

With 95% confidence, the confidence interval is $-0.1716763 \text{ to } -0.08994155$

Both statistics indicate that most computer scientists are satisfied with their jobs. However, the fact that the overall satisfaction rating is higher than the average of the different factors allows us to draw some interesting speculations. In particular, computer scientists may find their jobs to be greater than the sum of its parts.

### 2.4 Parents’ Educational Background and Salary

We wanted to use linear regression to see if there was some kind of relationship between the education level of the parents of a graduate and their salary. Our initial hypothesis was that someone who has more educated parents would likely be earning more due to the presumption that the more educated parents are able to provide more opportunities for their children.

Where the education level is:
<table>
<thead>
<tr>
<th>Number</th>
<th>Education level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Less Than High School</td>
</tr>
<tr>
<td>2</td>
<td>High School Diploma</td>
</tr>
<tr>
<td>3</td>
<td>Some College</td>
</tr>
<tr>
<td>4</td>
<td>Bachelor’s Degree</td>
</tr>
<tr>
<td>5</td>
<td>Master’s Degree</td>
</tr>
<tr>
<td>6</td>
<td>Professional Degree</td>
</tr>
<tr>
<td>7</td>
<td>Doctorate</td>
</tr>
</tbody>
</table>

Our findings are as follows.

As we can see from the graph, it appears that graduates who have more educated parents have higher salaries. We feel that this can be explained by the idea that more educated parents can give their children early access to tools that may help them later on in life. There are many factors to take into consideration that might help explain this relationship between parents education level and salaries.

We decided to go further and break down to Mother and Father.
Minimum 77182.92
Maximum 80990.75

Minimum 75121.9
Maximum 83824.19
Interestingly, graduates with well educated mothers don’t seem to have salaries significantly higher than those with less educated mothers. On the other hand, it appears as though graduates with more educated fathers are likely to earn more than those with less educated fathers. Again, there are many social and economical factors that may explain this observation. Note that we want to emphasize that salary is not the only measure of success. There have been many admirable people, and peers, in the field of computer science who, despite having less educated parents, have been and are very successful.

3 Amazon Product Links

This data shows the relationships between products that become evident when observing consumer buying patterns. Analyzing this kind of data has been key to Amazon’s success; once an accurate prediction can be made about a product, the retailer can customize the online shopping experience for the consumer. Aside from some ethical issues, data like this has much practical significance for both the consumer and retailer. Note that, in the data from STAN, a product i being frequently purchased with j does not necessarily indicate the same for product j with i.

3.1 Confidence Interval for Mean of Degree Distribution

Out of 262111 products (nodes) with 1234877 edges (co-purchasing links), we found the following data. Code is in the Appendix.

<table>
<thead>
<tr>
<th>Mean</th>
<th>4.794336</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Deviation</td>
<td>0.7233048</td>
</tr>
</tbody>
</table>

With 95% confidence, the confidence interval for the product links ranges from 4.791542 to 4.797129.

3.2 Proportion of outlinks not reciprocated

This is the confidence interval for the proportion of outlinks that are not reciprocated. Meaning that we only take into account one-way links.

<table>
<thead>
<tr>
<th>Number of outlinks NOT reciprocated</th>
<th>564707</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of outlinks</td>
<td>1234877</td>
</tr>
<tr>
<td>Proportion</td>
<td>0.457 = 45.7%</td>
</tr>
</tbody>
</table>

With 95% confidence, the confidence interval for the proportion of outlinks that are not reciprocated is from 0.4559988 to 0.4585975.

According to our analysis, almost half of the outlinks that products have are reciprocated. This proportion is an amazing indication of Amazon’s ability to market the right products to the right consumers. The consumer is appropriately introduced to products they might not have otherwise found, and Amazon’s sales are bolstered, so in a way, everyone reaps the benefits. With its sophisticated targeted system of e-commerce, there is no doubt why Amazon is in the forefront of online sales.
4 Summary

By observing data from the National Survey of Recent College Graduates (from 2003), we were able to draw some very interesting conclusions. There are many more, seemingly infinite, combinations of data points we could have observed, which would have revealed some social phenomena. From the Amazon product links, we saw just a piece of how an online retailer was able to weather the dot-com crash, and further become the largest online retailer. The number of interpretations we can draw from raw data is breathtaking. If we widen our scope from just college graduates and product associations to many other areas, it is startling to imagine the kind of possibilities.

5 Credits

Alex wrote the initial parsing code to import the raw data into R. Everyone helped write and debug code for problem 2, and each part of Problem 1, with each member having a slight focus on each part. Rajiv did the writeup in \LaTeX. We are happy to say that everyone in group worked together well and contributed equally to this project and over the course of the quarter.
A  Graphs of Salary Importance and Satisfaction Densities

Importance given to Salary.

Computer Science

Frequency

Importance given to salary
Actual Satisfaction with Salary on the job.

**Computer Science**

<table>
<thead>
<tr>
<th>Satisfaction of salary</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>2.0</td>
<td>1000</td>
</tr>
<tr>
<td>2.5</td>
<td></td>
</tr>
<tr>
<td>3.0</td>
<td>400</td>
</tr>
<tr>
<td>3.5</td>
<td></td>
</tr>
<tr>
<td>4.0</td>
<td></td>
</tr>
</tbody>
</table>

**Economics**

<table>
<thead>
<tr>
<th>Satisfaction of salary</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>2.0</td>
<td>700</td>
</tr>
<tr>
<td>2.5</td>
<td></td>
</tr>
<tr>
<td>3.0</td>
<td>400</td>
</tr>
<tr>
<td>3.5</td>
<td></td>
</tr>
<tr>
<td>4.0</td>
<td></td>
</tr>
</tbody>
</table>
B Code

B.1 Problem 1A

```r
probl_a_init <- function() {
  SATSAL_CS <<- NULL
  SATSAL_ECON <<- NULL
  SATSAL_COMM <<- NULL

  FACSAL_CS <<- NULL
  FACSAL_ECON <<- NULL
  FACSAL_COMM <<- NULL

  satSalCsSD <<- NULL
  facSalCsSD <<- NULL
  satSalEconSD <<- NULL
  facSalEconSD <<- NULL
  satSalCommSD <<- NULL
  facSalCommSD <<- NULL

  satSalCsMean <<- NULL
  facSalCsMean <<- NULL
  satSalEconMean <<- NULL
  facSalEconMean <<- NULL
  satSalCommMean <<- NULL
  facSalCommMean <<- NULL

  csmajor <<- c(116730)
}
```
econmajor <- c(419230)
comm <- c(766610)

probl_readinput <- function() {
  # Open file for reading
  con <- file('ECG03.DAT')
  open(con)

  # We parse through the file and add the response of salary satisfaction
  # and salary expectation to their respective majors
  while(length(myline <- readLines(con, n=1, warn=FALSE)) > 0) {
    # split and unlist each line so that we can access the elements of the line as
    # a vector
    myline <- noquote(unlist(strsplit(myline, ' ')))
    mmajor <- paste(myline[575:580], collapse=' ')

    if(mmajor==csmajor) {
      SATSAL_CS <<- c(SATSAL_CS, myline[926])
      FACSAL_CS <<- c(FACSAL_CS, myline[384])
    }
    if(mmajor==econmajor) {
      SATSAL_ECON <<- c(SATSAL_ECON, myline[926])
      FACSAL_ECON <<- c(FACSAL_ECON, myline[384])
    }
    if(mmajor==comm) {
      SATSAL_COMM <<- c(SATSAL_COMM, myline[926])
      FACSAL_COMM <<- c(FACSAL_COMM, myline[384])
    }
  }
  close(con)

  # For each major replace a value 'L' (not answered) with NA
  SATSAL_CS <<- noquote(SATSAL_CS)
  SATSAL_CS[SATSAL_CS=='L'] <<- NA
  SATSAL_CS <<- as.integer(SATSAL_CS)

  FACSAL_CS <<- noquote(FACSAL_CS)
  FACSAL_CS[FACSAL_CS=='L'] <<- NA
  FACSAL_CS <<- as.integer(FACSAL_CS)

  SATSAL_ECON <<- noquote(SATSAL_ECON)
  SATSAL_ECON[SATSAL_ECON=='L'] <<- NA
  SATSAL_ECON <<- as.integer(SATSAL_ECON)

  FACSAL_ECON <<- noquote(FACSAL_ECON)
  FACSAL_ECON[FACSAL_ECON=='L'] <<- NA
  FACSAL_ECON <<- as.integer(FACSAL_ECON)

  SATSAL_COMM <<- noquote(SATSAL_COMM)
  SATSAL_COMM[SATSAL_COMM=='L'] <<- NA
  SATSAL_COMM <<- as.integer(SATSAL_COMM)

  FACSAL_COMM <<- noquote(FACSAL_COMM)
  FACSAL_COMM[FACSAL_COMM=='L'] <<- NA
  FACSAL_COMM <<- as.integer(FACSAL_COMM)
}
prob1_mean <- function() {
  # This function calculates and outputs the means
  satSalCsMean <- mean(SATSAL_CS, na.rm=T)
  facSalCsMean <- mean(FACSAL_CS, na.rm=T)
  satSalEconMean <- mean(SATSAL_ECON, na.rm=T)
  facSalEconMean <- mean(FACSAL_ECON, na.rm=T)
  satSalCommMean <- mean(SATSAL_COMM, na.rm=T)
  facSalCommMean <- mean(FACSAL_COMM, na.rm=T)

  cat("mean of SATSAL_CS: ", satSalCsMean, 
  cat("mean of SATSAL_ECON: ", satSalEconMean, 
  cat("mean of SATSAL_COMM: ", satSalCommMean, 
  cat("mean of FACSAL_CS: ", facSalCsMean, 
  cat("mean of FACSAL_ECON: ", facSalEconMean, 
  cat("mean of FACSAL_COMM: ", facSalCommMean, 

  jpeg('images/SalarySatisfactionGraph.jpg')

  barplot(ylim=c(1.5,2.0), offset=1.5, c(satSalCsMean, satSalEconMean, satSalCommMean) -1.5, xlab="Majors", ylab="Average Satisfaction", names.arg=c("CS", "Econ", "Comm"))

  dev.off()
  jpeg('images/SalaryExpectationGraph.jpg')

  barplot(ylim=c(1.00,1.75), offset=1.00, c(facSalCsMean, facSalEconMean, facSalCommMean) -1.00, xlab="Majors", ylab="Average Importance", names.arg=c("CS", "Econ", "Comm"))

  dev.off()
}

prob1_sd <- function() {
  # Here we calculate the standard deviation using built-in R functions
  satSalCsSD <- sd(SATSAL_CS, na.rm=T)
  facSalCsSD <- sd(FACSAL_CS, na.rm=T)
  satSalEconSD <- sd(SATSAL_ECON, na.rm=T)
  facSalEconSD <- sd(FACSAL_ECON, na.rm=T)
  satSalCommSD <- sd(SATSAL_COMM, na.rm=T)
  facSalCommSD <- sd(FACSAL_COMM, na.rm=T)

  cat("sd of SATSAL_CS: ", satSalCsSD, 
  cat("sd of SATSAL_ECON: ", satSalEconSD, 
  cat("sd of SATSAL_COMM: ", satSalCommSD, 
  cat("sd of FACSAL_CS: ", facSalCsSD, 
  cat("sd of FACSAL_ECON: ", facSalEconSD, 
  cat("sd of FACSAL_COMM: ", facSalCommSD, 

}

prob1_mean_CI <- function() {
  # Calculate confidence interval for the difference between the two majors.
  # CS and Econ
B.2 Problem 1B

Very similar Problem 1A, only major changes are shown here.

```r
prob1b_proportion <- function() {
  # Table provides the counts of each type of element, organized alphabetically.
  countsCS <- table(PRO_MEETINGS_CS)
  cat("Proportion of CS majors who have attended professional meetings in the past year is: ", proportionCS, "\n")

  countsEcon <- table(PRO_MEETINGS_ECON)
  proportionEcon <- countsEcon[2]/(countsEcon[1] + countsEcon[2])
  cat("Proportion of Econ majors who have attended professional meetings in the past year is: ", proportionEcon, "\n")

  countsComm <- table(PRO_MEETINGS_COMM)
  cat("Proportion of Econ majors who have attended professional meetings in the past year is: ", proportionComm, "\n")
}
```
B.3 Problem 1C

Only major changes from 1A

```r
prob1b_CI <- function() {
  n1 = table(PRO_MEETINGS_CS) [1] + table(PRO_MEETINGS_CS) [2]
  n2 = table(PRO_MEETINGS_ECON) [1] + table(PRO_MEETINGS_ECON) [2]
  n3 = table(PRO_MEETINGS_COMM) [1] + table(PRO_MEETINGS_COMM) [2]
  SD1 = sqrt(proportionCS*(1-proportionCS)/n1)
  SD2 = sqrt(proportionEcon*(1-proportionEcon)/n2)
  SD3 = sqrt(proportionComm*(1-proportionComm)/n3)
  CI_CS_ECON = sqrt((SD1^2)/n1 + (SD2^2)/n2)
  CI_CS_COMM = sqrt((SD1^2)/n1 + (SD3^2)/n3)
  cat("Confidence interval for the difference between CS and Econ majors: (", proportionCS - proportionEcon - CI_CS_ECON, ",", proportionCS + CI_CS_ECON, ") \n")
  cat("Confidence interval for the difference between CS and Comm majors: (", proportionCS - proportionComm - CI_CS_COMM, ",", proportionCS + CI_CS_COMM, ") \n")
}

prob1_mean <- function() {
  satavgsumCsMean <<- mean(SATAVGSUM_CS, na.rm=TRUE)
  satoverallCsMean <<- mean(SATOVERALL_CS, na.rm=TRUE)
  cat("mean of satavgsum_cs: ", satavgsumCsMean, "\n")
  cat("mean of satoverall_cs: ", satoverallCsMean, "\n")
  jpeg("images/Prob1c.jpg")
  barplot(ylim=c(1.50, 1.80), offset=1.50, c(satavgsumCsMean, satoverallCsMean)
           -1.50, names.arg=c("Avg Sum", "Overall"))
  dev.off()
}

prob1_sd <- function() {
  satavgsumCsSD <<- sd(SATAVGSUM_CS, na.rm=TRUE)
  satoverallCsSD <<- sd(SATOVERALL_CS, na.rm=TRUE)
  cat("sd of satavgsum_cs: ", satavgsumCsSD, "\n")
  cat("sd of satoverall_cs: ", satoverallCsSD, "\n")
}

prob1_mean_CI <- function() {
  MoE1 <- 1.96 * satavgsumCsSD/sqrt(length(SATAVGSUM_CS))
  MoE2 <- 1.96 * satoverallCsSD/sqrt(length(SATOVERALL_CS))
  cat("The confidence interval for satisfaction of CS is: (", satavgsumCsMean - MoE1, ",", satavgsumCsMean + MoE1, ") \n")
  cat("The confidence interval for overall satisfaction of CS is: (", satoverallCsMean - MoE2, ",", satoverallCsMean + MoE2, ") \n")
}
```
B.4 Problem 1D

```r
prob1_readinput <- function() {
  con <- file('ECG03.DAT')
  open(con)

  while(length(myline <- readLines(con, n=1, warn=FALSE)) > 0) {
    myline <- noquote(unlist(strsplit(myline, ' ')))

    mymajor <- paste(myline[575:580], collapse='')
    if(mymajor==csmajor) {
      edu_father <- myline[356]
      edu_mother <- myline[357]
      edu_father <- as.integer(edu_father)
      edu_mother <- as.integer(edu_mother)
      edu_father[edu_father == '8'] <- NA
      edu_mother[edu_mother == '8'] <- NA
      salary <- paste(myline[906:912], collapse='')
      salary <- as.integer(salary)
      salary[salary == '999998'] <- NA
      salary[edu_father == '8'] <- NA
      salary[edu_mother == '8'] <- NA
      FATHER_EDU_CS <- c(FATHER_EDU_CS, edu_father)
      MOTHER_EDU_CS <- c(MOTHER_EDU_CS, edu_mother)
      SALARY <- c(SALARY, salary)
    }
  }
  close(con)
}

prob1_regress <- function() {
  md <- matrix(nrow=length(SALARY), ncol=3)
  md[,1] <- FATHER_EDU_CS
  md[,2] <- MOTHER_EDU_CS
  md[,3] <- SALARY

  i <- 1
  for(a in md[,3]) {
    if(is.na(md[i,1] + md[i,2] + md[i,3])) {
      md[i,3] <- NA
      md[i,2] <- NA
      md[i,1] <- NA
    }
    i <- i + 1
  }
  md <- na.omit(md)

  momAndDad <- md[,1] + md[,2]
  lmout <- lm(md[,3] ~ momAndDad)
  lmoutMom <- lm(md[,3] ~ md[,2])
  lmoutDad <- lm(md[,3] ~ md[,1])
  jpeg("images/MomEduSalary.jpg")
  plot(md[,2], md[,3], xlab="Education level of Mother", ylab="Salary")
  abline(lmoutMom)
  dev.off()
}
```
B.5 Problem 2

```r
prob2_init <- function() {
  myline <- NULL
  amazon <- NULL
  outdegrees <- NULL
  uberlist <- list(list())
}

read_input <- function() {
  con <- file('amazon0302.txt')
  open(con)

  myline <- readLines(con, n=4, warn=FALSE)
  myline <- readLines(con, warn=FALSE)
  myline <- as.integer(noquote(unlist(strsplit(myline, '\t'))))
  amazon <- matrix(nrow = (length(myline)/2), ncol = 2)
  oddrows <- seq(1, (dim(amazon)[1])*2, by=2)
  amazon[,1] <- myline[oddrows]

  close(con)
}

prob2a <- function() {
  # Storing the outdegree frequencies
  outdegrees <- table(amazon[,1])

  # Calculate the mean, standard deviation, and interval.
  EV <- mean(outdegrees)
  SD <- sd(outdegrees)
  MoE <- 1.96 * SD/sqrt(length(outdegrees))

  cat("The 95% confidence interval for the product links is: (", EV - MoE, ",", EV + MoE, ") \n")
}
```

prob2b <- function()
{
  count <- 0

  for ( i in 1:(length(myline)/2)){
    uberlist [[1]][[(amazon[i,1]+1)]] <<- NA
  }

  for ( i in 1:(length(myline)/2)){
    uberlist [[1]][[(amazon[i,1]+1)]] <<- append(uberlist [[1]][[(amazon[i,1]+1)]] , (amazon[i,2]+1))
  }

  for ( i in 1:(length(myline)/2)){
    uberlist [[1]][[(amazon[i,1]+1)]] <<- as.vector(na.omit(uberlist [[1]][[(amazon[i ,1]+1)]]))
  }
}