OkCupid Data for Introductory Statistics and Data Science Courses

Albert Y. Kim *
Department of Mathematics
Middlebury College, Middlebury, VT

Adriana Escobedo-Land
Environmental Studies-Biology Program
Reed College, Portland, OR

May 29, 2015

*Address for correspondence: Department of Mathematics, Middlebury College, Warner Hall, 303 College Street, Middlebury, VT 05753. Email: aykim@middlebury.edu.

OkCupid Data for Introductory Statistics and Data Science Courses

Abstract

We present a data set consisting of user profile data for 59,946 San Francisco OkCupid users (a free online dating website) from June 2012. The data set includes typical user information, lifestyle variables, and text responses to 10 essays questions. We present four example analyses suitable for use in undergraduate introductory probability and statistics and data science courses that use R. The statistical and data science concepts covered include basic data visualization, exploratory data analyses, multivariate relationships, text analysis, and logistic regression for prediction.

Keywords: OkCupid, online dating, data science, big data, logistic regression, text mining.

1 Introduction

Given that the field of data science is gaining more prominence in academia and industry, many statisticians are arguing that statistics needs to stake a bigger claim in data science in order to avoid marginalization by other disciplines such as computer science and computer engineering (Davidson, 2013; Yu, 2014). The importance of emphasizing data science concepts in the undergraduate curriculum is stressed in the American Statistical Association's (ASA) most recent Curriculum Guidelines for Undergraduate Programs in Statistical Science (American Statistical Association Undergraduate Guidelines Workgroup, 2014).

While precise definition of the exact difference between statistics and data science and its implications for statistics education can be debated (Wickham, 2014), one consensus among many in statistics education circles is that at the very least statistics needs to incorporate a heavier computing component and increase the use of technology for both developing conceptual understanding and analyzing data (GAISE College Group, 2005; Nolan and Lang, 2010). Relatedly, in the hopes of making introductory undergraduate statistics courses more relevant, many statistics educators are placing a higher emphasis on the use of real data in the classroom, a practice the ASA's Guidelines for Assessment and Instruction in Statistics Education (GAISE) project's report strongly encourages (GAISE College Group, 2005). Of particular importance to the success of such ambitions are the data sets considered, as they provide the context of the analyses and thus will ultimately drive student interest (Gould, 2010).

It is in light of these discussions that we present this paper centering on data from the online dating website OkCupid, specifically a snapshot of San Francisco California users taken on June 2012. We describe the data set and present a series of example analyses along with corresponding pedagogical discussions. The example analyses presented in this paper were used in a variety of settings at Reed College in Portland, Oregon: a 90 minute introductory tutorial on R, an introductory probability and statistics course, and a follow-up two-hundred level data science course titled "Case Studies in Statistical Analysis." The statistical and data science concepts covered include basic data visualization, exploratory data analyses, multivariate relationships, text analysis, and logistic regression for prediction. All examples are presented using the R statistical software program and make use of the `mosaic`, `dplyr`, `stringr`, and `ggplot2` packages (Wickham, 2009, 2012; Pruim et al., 2014; Wickham, 2012; Wickham and Francois, 2014).

2 Data

The data consists of the public profiles of 59,946 OkCupid users who were living within 25 miles of San Francisco, had active profiles on June 26, 2012, were online in the previous year, and had at least one picture in their profile. Using a Python script, data was scraped from users' public profiles on June 30, 2012; any non-publicly facing information such as messaging was not accessible.

Variables include typical user information (such as sex, sexual orientation, age, and ethnicity) and lifestyle variables (such as diet, drinking habits, smoking habits). Furthermore, text responses to the 10 essay questions posed to all OkCupid users are included as well, such as "My Self Summary," "The first thing people usually notice about me...," and "On a typical Friday night I am..." For a complete list of variables and more details, see the accompanying codebook `okcupid_codebook.txt`. We load the data as follows:

```
profiles <- read.csv(file="profiles.csv", header=TRUE, stringsAsFactors=FALSE)
n <- nrow(profiles)
```

Analyses of similar data has received much press of late, including Amy Webb's TED talk "How I Hacked Online Dating" (Webb, 2013) and Wired magazine's "How a Math Genius Hacked OkCupid to Find True Love" (Poulsen, 2014). OkCupid co-founder Christian Rudder pens periodical analyses of OkCupid data on the blog OkTrends (`http://blog.okcupid.com/`) and has recently published a book "Dataclysm: Who We Are When We Think No One's Looking" describing similar analyses (Rudder, 2014). Such publicity surrounding data-driven online dating and the salience of dating matters among students makes this data set one with much potential to be of interest to students, hence facilitating the instruction of statistical and data science concepts.

Before we continue we note that even though this data consists of publicly facing material, one should proceed with caution before scraping and using data in fashion similar to ours, as the Computer Fraud and Abuse Act (CFAA) makes it a federal crime to access a computer without authorization from the owner (Penenberg, 2014). In our case, permission to use and disseminate the data was given by its owners (See Acknowledgements).

3 Example Analyses

We present example analyses that address the following questions:

1. How do the heights of male and female OkCupid users compare?
2. What does the San Francisco online dating landscape look like? Or more specifically, what is the relationship between users' sex and sexual orientation?
3. Are there differences between the sexes in what words are used in the responses to the 10 essay questions?
4. How accurately can we predict a user's sex using their listed height?

For each question, we present an exercise as would be given to students in a lab setting, followed by a pedagogical discussion.

3.1 Male and Female Heights

3.1.1 Exercise

We compare the distribution of male and female OkCupid users' heights. Height is one of 3 numerical variables in this data set (the others being age and income). This provides us an opportunity to investigate numerical summaries using the `favstats()` function from the `mosaic` package:

```
require(mosaic)
favstats(height, data=profiles)

##  min Q1 median Q3 max mean sd     n missing
##    1 66     68 71  95   68  4 59943       3
```

We observe that some of the heights are nonsensical, including heights of 1 inch and 95 inches (equaling 7'11"). We deem heights between 55 and 80 inches to be reasonable and remove the rest. While there is potential bias in discarding users with what we deem non-reasonable heights, since out of the 59946 users there are only 117 who would be discarded, the effect would not be substantial. Therefore we keep only those users with heights between 55 and 80 inches using the `filter()` function from the `dplyr` package:

```
require(dplyr)
profiles.subset <- filter(profiles, height>=55 & height <=80)
```

We compare the distributions of male and female heights using histograms. While we could plot two separate histograms without regard to the scale of the two axes, in Figure 1 we instead use the `histogram()` function from the `mosaic` package to:

1. Plot heights given sex by defining the formula: `~ height | sex`.

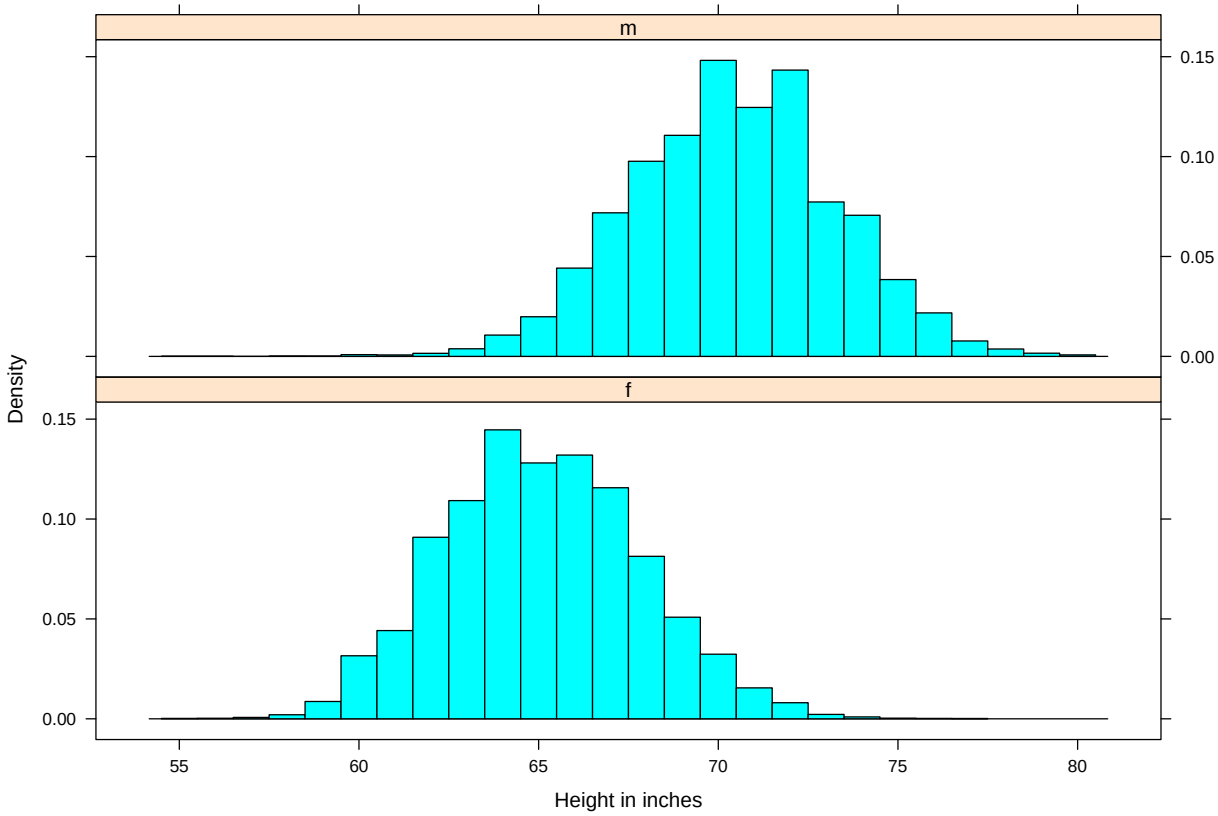

Figure 1: Histograms of user heights split by sex.

2. Plot them simultaneously in a *lattice* consisting of two rows and one column of plots by setting `layout=c(1,2)`

3. Plot them with bin widths matching the granularity of the observations (inches) by setting `width=1`. The `histogram()` function automatically matches the scales of the axes for both plots.

```
histogram(~height | sex, width=1, layout=c(1,2), xlab="Height in inches",
          data=profiles.subset)
```

3.1.2 Pedagogical Discussion

This first exercise stresses many important considerations students should keep in mind when working with real data. Firstly, it emphasizes the importance of performing an exploratory data analysis to identify anomalous observations and confronts students with the question of what to do with them. For example, while a height of 1 inch is clearly an outlier that needs to be removed, at what point does a height no longer become reasonable and what impact does the removal of unreasonable heights have on the conclusions? In our case, since only a small number of observations are removed, the impact is minimal.

Secondly, this exercise demonstrates the power of simple data visualizations such as histograms to convey insight and hence emphasizes the importance of putting careful thought into their construction. In our case,

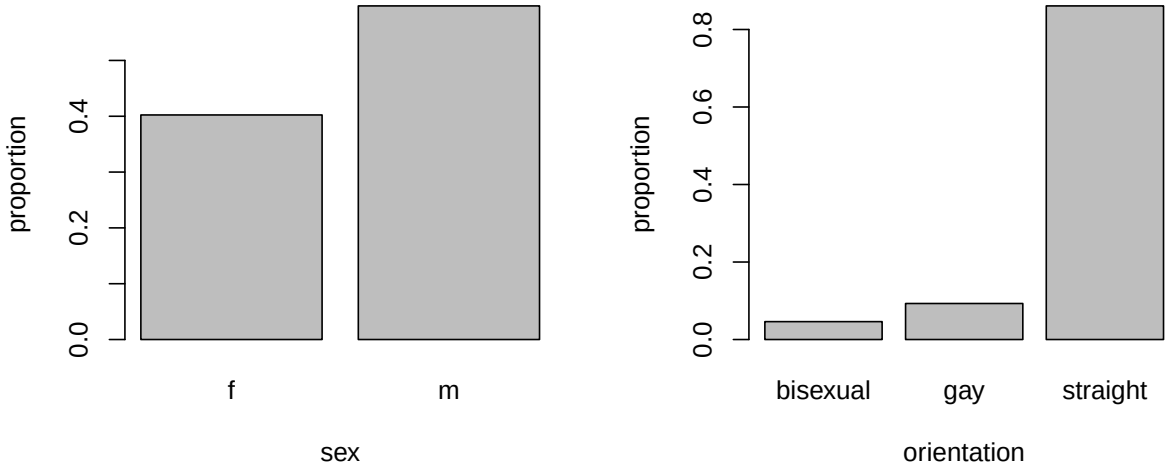

Figure 2: Distributions of sex and sexual orientation.

while having students plot two histograms simultaneously in order to demonstrate that males have on average greater height may seem to be a pedantic goal at first, we encouraged students to take a closer look at the histograms and steered their focus towards the unusual peaks at 72 inches (6 feet) for males and 64 inches (5'4") for females. Many of the students could explain the phenomena of the peak at 72 inches for men: sociological perceptions of the rounded height of 6 feet. On the other hand, consensus was not as strong about perceptions of the height of 5'4" for women. Instructors can then refer students to the entry on OkCupid's blog OkTrends "The Biggest Lies in Online Data" (Rudder, 2010) to show they have replicated (on a smaller scale) a previous analysis and then show other analyses conducted by OkCupid.

Further questions that can be pursued from this exercise include "How can we question if those peaks are significant or due to chance?," "Are we only observing men who are just under 6 feet rounding up, or are men just over 6 feet rounding down as well?," or "How can we compare the distribution of listed heights on OkCupid to the actual San Francisco population's heights?"

3.2 Relationship Between Sex and Sexual Orientation

3.2.1 Exercise

Since among the most important considerations in assessing a potential mate are their sex and sexual orientation, in this exercise we investigate the relationship between these two variables. At the time, OkCupid allowed for two possible sex choices (male or female) and three possible sexual orientation choices (gay, bisexual, or straight)[1]. First, we perform a basic exploratory data analysis on these variables using barcharts in Figure 2:

```
par(mfrow=c(1, 2))
barplot(table(profiles$sex)/n, xlab="sex", ylab="proportion")
barplot(table(profiles$orientation)/n, xlab="orientation", ylab="proportion")
```

[1] OkCupid has since relaxed these categorizations to allow for a broader range of choices for both sex and sexual orientation.

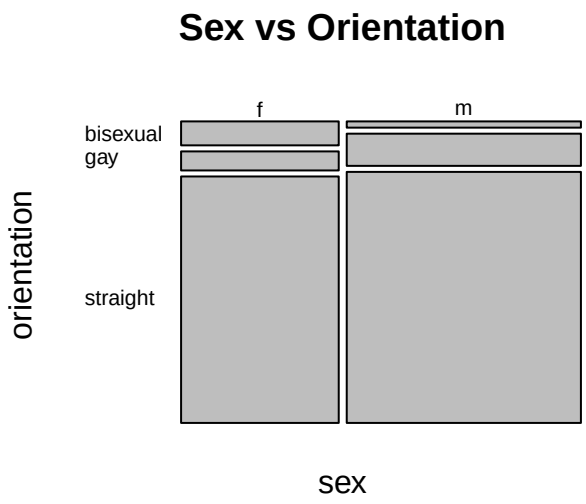

Figure 3: Joint distribution of sex and sexual orientation.

However, in order to accurately portray the dating landscape we can't just consider the **marginal distributions** of these variables, we must consider their **joint** and **conditional distributions** i.e. the cross-classification of the two variables. We describe the distribution of sexual orientation conditional on sex. For example, we can ask of the female population, what proportion are bisexual? We do this using the `tally()` function from the `mosaic` package and ensure both columns sum to 1 by setting `format='proportion'`. Furthermore, we visualize their joint distribution, as represented by their contingency table, via the mosaicplot shown in Figure 3.

```
tally(orientation ~ sex, data=profiles, format='proportion')
```

```
##             sex
## orientation     f     m
##    bisexual 0.083 0.022
##    gay      0.066 0.111
##    straight 0.851 0.867
```

```
sex.by.orientation <- tally(~sex + orientation, data=profiles)
sex.by.orientation
```

```
##    orientation
## sex bisexual   gay straight
##   f     1996  1588    20533
##   m      771  3985    31073
```

```
mosaicplot(sex.by.orientation, main="Sex vs Orientation", las=1)
```

Do these results generalize to the entire San Francisco online dating population?

3.2.2 Pedagogical Discussion

This exercise is an opportunity to reinforce statistical notions such as marginal/joint/conditional distributions and sampling bias. The data indicate that the San Francisco OkCupid dating population skews male and while the proportions of males and females who list themselves as straight are similar, a higher proportion of males list themselves as gay while a higher proportion of females list themselves as bisexual. Many students were not surprised by these facts as they were well aware of the gender imbalance issues in the large technology sector in the San Francisco Bay Area and San Francisco's history of being a bastion for the gay community.

The question of generalizability was presented in an introductory probability and statistics assignment. Almost all students were able to recognize the selection biases of who signs up for this particular site and hence the non-generalizability of the results. For example, some recognized that OkCupid's demographic is most likely different than other dating websites' demographics such as `match.com` (which is not free) or `christiansingles.com` (which is targeted towards Christians). So while 59946 users may initially seem like a large sample, we emphasized to students that bigger isn't always better when it comes to obtaining accurate inference. This proved an excellent segue to Kate Crawford of Microsoft Research's YouTube talk "Algorithmic Illusions: Hidden Biases of Big Data" (Crawford, 2013) where she discusses examples of sampling bias in the era of "Big Data."

Further questions one can pose to students include "Which dating demographic would you say has it the best and worst in terms of our simplified categorization?" and "What variable do you think should be incorporated next in order to represent the OkCupid dating pool as faithfully as possible?"

3.3 Text Analysis

3.3.1 Exercise

The next exercise focuses on the responses to the essay questions, providing an opportunity to perform text analysis. Words are called "strings" in the context of computer programming. Manipulating text data in R is often a complicated affair, so we present some code that is at an intermediate level to preprocess the essay responses for analysis. The following code outputs a single vector `essays` that contains all 10 essay responses for each user concatenated together:

- We use the `select()` function from the `dplyr` package to select the 10 essay columns as identified by the fact they `starts_with("essay")`.
- For each user, we concatenate the 10 columns to form a single character string. The code applies the function `paste(x, collapse=" ")` to every essay, where `x` is a user's set of 10 essay responses and the `paste()` function collapses `x` across columns while separating the elements by a space. We do this for each set of essays (i.e. each row of `essays`) via the `apply()` function and with the `MARGIN` argument set to `1`.
- We replace all HTML line breaks (`\n`) and paragraph breaks (`<br />`) with spaces using the `str_replace_all()` function from the `stringr` package to make the outputs more readable.

```
require(stringr)
essays <- select(profiles, starts_with("essay"))
essays <- apply(essays, MARGIN=1, FUN=paste, collapse=" ")
essays <- str_replace_all(essays, "\n", " ")
essays <- str_replace_all(essays, "<br />", " ")
```

We ask: Do male and female OkCupid users use words at different rates in their essay responses? We search for the presence of a word in a user's essays using the `str_detect()` function in the `stringr` package. We then use the `tally()` function illustrated in Section 3.2 to compute the distribution conditional on sex. For example, the word "book" is used in 62% of female profiles and 55% of male profiles:

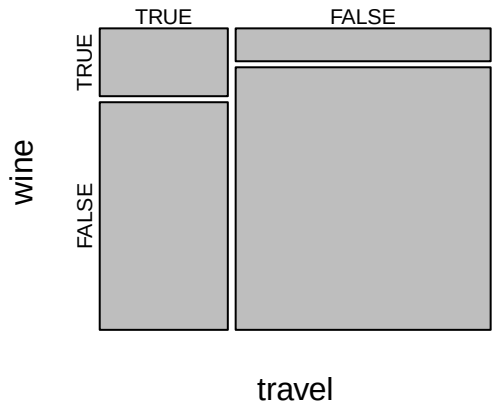

Figure 4: Co-occurrence of 'travel' and 'wine.'

```
profiles$has.book <- str_detect(essays, "book")
tally(has.book ~ sex, profiles, format='proportion')

##         sex
## has.book    f    m
##    TRUE  0.62 0.55
##    FALSE 0.38 0.45
```

In Table 1, we make similar comparisons for the use of the words "travel," "food," "wine," and "beer."

word	female	male
travel	0.386	0.299
food	0.652	0.601
wine	0.201	0.117
beer	0.087	0.109

Table 1: Proportions of each sex using word in essays.

We further study the co-occurrence of words, such as "wine" and "travel," visualizing their relationship in a mosaicplot in Figure 4.

```
profiles$has.wine <- str_detect(essays, "wine")
profiles$has.travel <- str_detect(essays, "travel")
travel.vs.wine <- tally(~has.travel + has.wine, data=profiles)
mosaicplot(travel.vs.wine, main="", xlab="travel", ylab="wine")
```

We can also evaluate the statistical significance of the difference in the use of the words, such as the word "football," via a two-sample proportions test using the `prop.test()` function: you specify the vectors `x` of

the successes of each group (the first row of `results`) and `n` of the number of observations in each group (the sums of the columns of `results`). While the difference of around 3.6% - 3.1% = 0.5% yields a p-value that is small, suggesting statistical significance, it can be argued that this difference is of little practical significance.

```
profiles$has.football <- str_detect(essays, "football")
results <- tally(~ has.football + sex, data=profiles)
prop.test(x=results[1, ], n=colSums(results), alternative="two.sided")

##
##  2-sample test for equality of proportions with continuity
##  correction
##
## data:  x and n
## X-squared = 13, df = 1, p-value = 0.0002929
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.0085 -0.0026
## sample estimates:
## prop 1 prop 2
##  0.031  0.036
```

And finally, consider the following fun exercise: we generate the top 500 words used by males and females respectively. The following code uses the "pipe" `%>%` operator from the `dplyr` package to use the output of one function as the first argument for the next function. For example, the following two lines of code perform the identical task:

```
c(1.1, 2.1, 3.1, 4.1) %>% sum() %>% round()
round(sum(c(1.1, 2.1, 3.1, 4.1)))
```

This allows us to avoid having multiple R functions nested in a large number of parentheses and highlights the functions used in a sequential fashion. In our case, the code below:

- Pulls the `subset` of `essays` corresponding to males (subsequently females).
- Splits up the each user's essay text at each space, i.e. cuts it up into words, using the `str_split()` function from the `stringr` package.
- Converts the list of words into a vector of words.
- Computes the frequency table using the `table()` function.
- Sorts them in decreasing order.
- Extracts the words (and not the frequency counts), which are the `names` of each element of the vector.

```
male.words <- subset(essays, profiles$sex == "m") %>%
  str_split(" ") %>%
  unlist() %>%
  table() %>%
  sort(decreasing=TRUE) %>%
  names()
female.words <- subset(essays, profiles$sex == "f") %>%
```

```
  str_split(" ") %>%
  unlist() %>%
  table() %>%
  sort(decreasing=TRUE) %>%
  names()
```

```
# Top 25 male words:
male.words[1:25]

##  [1] ""                "i"               "and"
##  [4] "the"             "to"              "a"
##  [7] "of"              "my"              "in"
## [10] "i'm"             "you"             "for"
## [13] "with"            "that"            "is"
## [16] "have"            "like"            "on"
## [19] "but"             "or"              "<a"
## [22] "at"              "class=\"ilink\"" "it"
## [25] "am"

# Top 25 female words
female.words[1:25]

##  [1] ""     "i"    "and"  "the"  "to"   "a"    "my"   "of"   "in"   "i'm"
## [11] "with" "for"  "you"  "that" "have" "is"   "love" "am"   "but"  "like"
## [21] "or"   "on"   "at"   "it"   "be"
```

However, for both males and females, the top words are not interesting in that they include many particles such as "I," "and," and "the" (see the top 25 below). Therefore, we consider the difference in words mentioned by males and similarly for females, by taking the difference in sets using the `setdiff()` function. Note that we didn't correct for punctuation.

```
# Words in the males top 500 that weren't in the females' top 500:
setdiff(male.words[1:500], female.words[1:500])

##  [1] ","         "video"     "company"   "sports"    "/"
##  [6] "internet"  "future"    "computer"  "star"      "well,"
## [11] "well."     "away"      "john"      "until"     "business"
## [16] "us"        "type"      "couple"    "generally" "2"
## [21] "more."     "went"      "bar"       "science"   "woman"
## [26] "work."     "started"   "does"      "here."     "found"
## [31] "three"     "lost"      "means"     "do."       "become"
## [36] "run"       "that,"

# Words in the male top 500 that weren't in the females' top 500:
setdiff(female.words[1:500], male.words[1:500])

##  [1] "loving"      "dancing,"    "love,"       "appreciate"  "dog"
##  [6] "hair"        "beautiful"   "laughing"    "passionate"  "red"
## [11] "cooking,"    ";)"          "laugh."      "please"      "kids"
## [16] "local"       "drinking"    "kinds"       "family."     "healthy"
## [21] "adventure"   "explore"     "laugh,"      "men"         "smile."
## [26] "comfortable" "crazy"       "nature"      "hiking,"     "day."
```

```
## [31] "chocolate"   "huge"        "change"      "dating"      "sex"
## [36] "met"         "movies."
```

3.3.2 Teaching Goals and Discussions

This exercise provides students with experience performing basic text processing, mining, and analysis. Given the more advanced tools used this exercise, we suggest this be reserved for students with more familiarity with R. We deliberately did not preprocess to remove punctuation and HTML tags, both to keep the code simple and to demonstrate the reality to students that "real" data is often very messy and requires work to clean up.

Statistical concepts covered include the difference between practical and statistical significance as demonstrated by the difference in proportion of males and females that used the word "football". This can lead to discussions of what it means to conduct hypothesis tests when the sample size is as large as 59946. Furthermore, we demonstrate that simple comparisons via basic set operations can be very powerful tools. For example, the difference in words used by males and females in our surface-level analysis is striking. The richness of the essay data allows students to verify and challenge prior sociological beliefs and preconceptions using empirical data.

Another interesting avenue for investigation is to what degree the above results hold when the comparison groups are further refined (grouping by sex *and* sexual orientation for example). Even bolder goals include introducing text analysis concepts such as regular expressions, inverse document frequency, natural language processing, and Latent Dirichlet Allocation (Blei et al., 2003).

3.4 Predictors of Sex

3.4.1 Exercise

This exercise provides an opportunity to fit a predictive model for sex using logistic regression. In order to reinforce the concepts of logistic regression, we keep things simple and consider only one predictor variable in the logistic model: height. We restrict consideration to users whose heights are "reasonable" as defined in Section 3.1 and in order to to speed up computation and improve graphical outputs, we only consider a random sample of 5995 users (10% of the data).

However, to ensure the replicability of our results (in other words ensuring the same 5995 users are "randomly" selected each time we run the code), we demonstrate the use of the `set.seed()` function. R's random number generator is not completely random, but rather is *pseudorandom* in that it generates values that are statistically indistinguishable from a truly random sequence of values, but are generated by a deterministic process. This deterministic process takes in a *seed* value and for the same seed value, R will generate the same sequence of values. For example, consider generating a random sequence of the numbers 1 through 10 using the `sample()` function for various seed values. We see that setting the seed to the same (arbitrarily chosen) value 76 yields the same sequence, whereas changing the seed value to 79 yields a difference sequence. Play around with this function to get a feel for it.

```
set.seed(76)
sample(1:10)
```

```
##  [1]  7 10  4  1  3  8  2  6  9  5
```

```
set.seed(76)
sample(1:10)
```

```
##  [1]  7 10  4  1  3  8  2  6  9  5
```

```
set.seed(79)
sample(1:10)
```

Figure 5: Female indicator vs height.

```
##  [1]  9  5  6 10  4  2  1  7  3  8
```

We proceed by setting the seed value to the value 76 and sample 5995 users at random by using the `sample_n()` function from the `dplyr` package

```
profiles <- filter(profiles, height>=55 & height <=80)
set.seed(76)
profiles <- sample_n(profiles, 5995)
```

We convert the `sex` variable to a binary `is.female` variable, whose value is `1` if the user is female and `0` if the user is male, using the `ifelse()` function. Alternatively, we could have coded `is.female` with `TRUE/FALSE` values, but for plotting purposes we code this variable using `1/0` numerical values. We create the `is.female` variable using the `mutate()` function from the `dplyr` package, which allows us create new variables from existing ones. We plot the points as in Figure 5, making use of the `ggplot2` package and defining an initial base plot.

```
require(ggplot2)
profiles <- mutate(profiles, is.female = ifelse(sex=="f", 1, 0))
base.plot <- ggplot(data=profiles, aes(x=height, y=is.female)) +
  scale_y_continuous(breaks=0:1) +
  theme(panel.grid.minor.y = element_blank()) +
  xlab("Height in inches") +
  ylab("Is female?")
```

We modify this base plot as we go:

```
base.plot + geom_point()
```

This plot is not very useful, as the overlap of the points makes it difficult for determine how many points are involved. We use the `geom_jitter()` function to add a little random noise to each clump of points both

Figure 6: Female indicator vs height (jittered).

along the x and y axes as shown in Figure 6. We observe, for example, there are much fewer males with heigh 63 inches than 70 inches.

```
base.plot + geom_jitter(position = position_jitter(width = .2, height=.2))
```

We fit both linear and logistic regression models using height as the sole predictor. In order to summarize the results, we use the `msummary()` function from the `mosaic` package rather than the standard `summary()` function, as its output is much more digestible. Furthermore, we extract the coefficients of the linear model using the `coef()` function.

```
linear.model <- lm(is.female ~ height, data=profiles)
msummary(linear.model)

##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.42887    0.08373    76.8   <2e-16 ***
## height      -0.08831    0.00122   -72.1   <2e-16 ***
##
## Residual standard error: 0.36 on 5993 degrees of freedom
## Multiple R-squared:  0.465,Adjusted R-squared:  0.464
## F-statistic: 5.2e+03 on 1 and 5993 DF,  p-value: <2e-16

b1 <- coef(linear.model)
b1

## (Intercept)      height
##       6.429      -0.088
```

```
logistic.model <- glm(is.female ~ height, family=binomial, data=profiles)
msummary(logistic.model)
```

```
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  44.9999     1.1374    39.6   <2e-16 ***
## height       -0.6705     0.0169   -39.8   <2e-16 ***
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8075.1  on 5994  degrees of freedom
## Residual deviance: 4460.2  on 5993  degrees of freedom
## AIC: 4464
##
## Number of Fisher Scoring iterations: 6

b2 <- coefficients(logistic.model)
b2

## (Intercept)      height
##       45.00       -0.67
```

In both cases, we observe that the coefficient associated with height is negative (-0.09 and -0.67 for the linear and logistic regressions respectively). In other words, as height increases, the fitted probability of being female decreases as is expected. We plot both regression lines in Figure 7, with the linear regression in red and the logistic regression in blue. The latter necessitates the function `inverse.logit()` in order to compute the inverse logit of the linear equation to obtain the fitted probabilities $\widehat{p}_i$:

$$\widehat{p}_i = \frac{\exp\left(\widehat{\beta}_0 + \widehat{\beta}_1 \times \text{height}_i\right)}{1 + \exp\left(\widehat{\beta}_0 + \widehat{\beta}_1 \times \text{height}_i\right)} = \frac{1}{1 + \exp\left(-(\widehat{\beta}_0 + \widehat{\beta}_1 \times \text{height}_i)\right)}$$

```
inverse.logit <- function(x, b){
  linear.equation <- b[1] + b[2]*x
  1/(1+exp(-linear.equation))
}
base.plot + geom_jitter(position = position_jitter(width = .2, height=.2)) +
  geom_abline(intercept=b1[1], slope=b1[2], col="red", size=2) +
  stat_function(fun = inverse.logit, args=list(b=b2), color="blue", size=2)
```

We observe that linear regression (red curve) yields fitted probabilities greater than 1 for heights less than 61 inches and less than 0 for heights over 73 inches, which do not make sense. This is not a problem with logistic regression as the shape of the logistic curve ensures that all fitted probabilities are between 0 and 1. We therefore deem logistic regression to be a more appropriate technique for this data than linear regression.

However, when predicting a user's gender, just using the fitted probabilities $\widehat{p}_i$ is insufficient; a decision threshold is necessary. In other words, a point at which if the fitted probability of a user being female is exceeded, we *predict* that user to be female. Looking at the histogram of fitted probabilities, we pick a decision threshold p^* such that for all users with $\widehat{p}_i > p^*$, we predict those users to be female. We opt for $p^* = 0.5$ since it splits the values somewhat nicely and highlight this value in red in Figure 8. In order to evaluate the performance of our model and our decision threshold, we produce a contingency table comparing the true (`is.female`) and predicted (`predicted.female`) values:

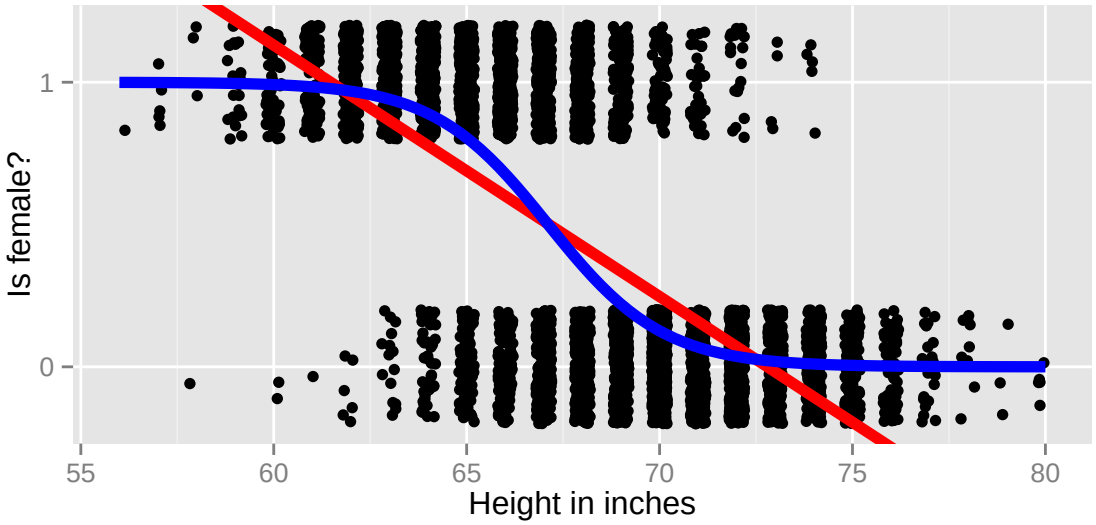

Figure 7: Predicted linear (red) and logistic (blue) regression curves.

```
profiles$p.hat <- fitted(logistic.model)
ggplot(data=profiles, aes(x=p.hat)) +
  geom_histogram(binwidth=0.1) +
  xlab(expression(hat(p))) +
  ylab("Frequency") +
  xlim(c(0,1)) +
  geom_vline(xintercept=0.5, col="red", size=1.2)
```

```
profiles <- mutate(profiles, predicted.female = p.hat >= 0.5)
tally(~is.female + predicted.female, data=profiles)

##          predicted.female
## is.female TRUE FALSE
##         0  566  3024
##         1 1953   452
```

How did our predictions fare?

3.4.2 Pedagogical Discussion

We find that the jump from linear to logistic regression is hard for many students to grasp at first. For example, students often ask "Why the log and exp functions?" and "So we are not modelling the outcome variable Y_i, we're modeling the probability p_i that Y_i equals 1?" This exercise allows students to build up to the notion of logistic regression from the ground up using visual tools. We also argue that on top of fitting models and interpreting any results, students should also use the results to make explicit predictions and evaluate any model's predictive power. We asked the students "For what proportion of people did the model guess wrong?" referring to the misclassification error rate, in this case 16.98%. Also solving for height using $p^* = 0.5$ yields a height of 67.11 inches, corresponding to 5 foot 7 inches, which is the height in Figure 1 at which the proportion of males starts to exceed the proportion of females. This point can be highlighted to students, tying together this exercise with the exercise in Section 3.1.

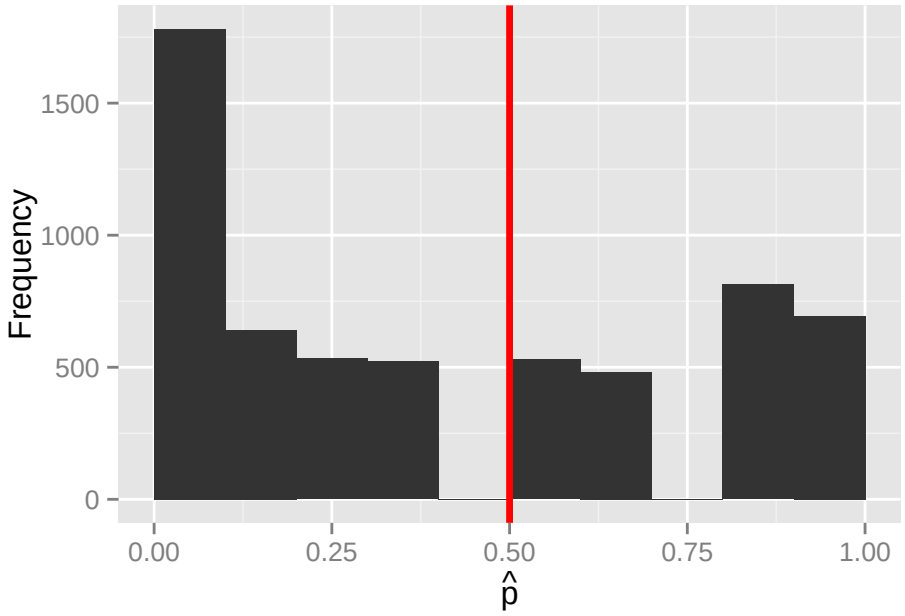

Figure 8: Fitted probabilities of being female and decision threshold (in red).

Further questions to ask of students include building a model with more than one predictor, incorporating essay information from Section 3.3, evaluating the *false positive rate* (the proportion of users who were predicted to be female who were actually male), evaluating the *false negative rate* (the proportion of users who were predicted to be male who were actually female), the impact of varying the decision threshold p^*, and asking questions about out-of-sample predictions (using different data to fit and evaluate the model).

4 Conclusions

We present a data set consisting of actual San Francisco OkCupid users' profiles in June 2012 and present example analyses of different levels of sophistication for direct use in the classroom in a similar fashion to Horton et al. (2015). We feel that this data set is ideal for use in introductory statistics and data science courses as the salience of the data set provides students with an interesting vehicle for learning important concepts. By presenting questions to students that allow for the use of their background knowledge, whether it be from the news, stereotypes, or sociological knowledge, students are much better primed to absorb statistical lessons. Furthermore,

1. The data consists of a rich array of categorical, ordinal, numerical, and text variables.
2. This is an instance of real data that is messy, has many suspicious values that need to be accounted for, and includes categorical variables of a complicated nature (for instance, there are 218 unique responses to the ethnicity variable). This reinforces to students that time and energy must be often invested into preparing data for analysis.
3. The data set is of modest size. While $n = 59946$ is not an overwhelmingly large number of observations, it is still much larger than typical data sets used in many introductory probability and statistics courses.

All the files, including the original data and the R Sweave `JSE.Rnw` file used to create this document, can be found at https://github.com/rudeboybert/JSE_OkCupid. Note that the file `profiles.csv.zip`

must be unzipped first. All R code used in this document can be outputted into an R script file by using the `purl()` function in the `knitr` package on `JSE.Rnw`:

```
library(knitr)
purl(input="JSE.Rnw", output="JSE.R", quiet=TRUE)
```

Acknowledgements

First, we thank OkCupid president and co-founder Christian Rudder for agreeing to our use of this data set (under the condition that the data set remains public). Second, we thank Everett Wetchler everett.wetchler@gmail.com for providing the data; the Python script used to scrape the data can be found at https://github.com/evee746/okcupid. Finally, we thank the reviewers for their helpful comments.

References

American Statistical Association Undergraduate Guidelines Workgroup (2014). "2014 Curriculum Guidelines for Undergraduate Programs in Statistical Science", *Technical report*, American Statistical Association, Alexandria, VA.
URL: *http://www.amstat.org/education/curriculumguidelines.cfm, last accessed April 11, 2015*

Blei, D. M., Ng, A. Y. and Jordan, M. I. (2003). "Latent Dirichlet Allocation", *Journal of Machine Learning Research* **3**: 993–1022.

Crawford, K. (2013). "Algorithmic Illusions: Hidden Biases of Big Data", *Strata Conference* .
URL: *https://www.youtube.com/watch?v=irP5RCdpilc, last accessed April 11, 2015*

Davidson, M. (2013). "Aren't We Data Science?", *AMSTAT News* .
URL: *http://magazine.amstat.org/blog/2013/07/01/datascience/, last accessed April 11, 2015*

GAISE College Group (2005). "Guidelines for Assessment and Instruction in Statistics Education", *Technical report*, American Statistical Association, Alexandria, VA.
URL: *http://www.amstat.org/education/gaise, last accessed April 11, 2015*

Gould, R. (2010). "Statistics and the Modern Student", *International Statistics Review* **78**(2): 297–315.

Horton, N. J., Baumer, B. and Wickham, H. (2015). "Setting the stage for data science: integration of data management skills in introductory and second courses in statistics", *CHANCE* **28**(2): 40–50.

Nolan, D. and Lang, D. T. (2010). "Computing in the Statistics Curricula", *The American Statistician* **64**(2): 97–107.

Penenberg, A. L. (2014). "Did the mathematician who hacked OkCupid violate federal computer laws?", *Pando Daily* .
URL: *http://pando.com/2014/01/22/did-the-mathematician-who-hacked-okcupid-violate-federal-computer-laws/, last accessed April 11, 2015*

Poulsen, K. (2014). "How a Math Genius Hacked OkCupid to Find True Love", *WIRED* .
URL: *http://www.wired.com/wiredscience/2014/01/how-to-hack-okcupid/, last accessed April 11, 2015*

Pruim, R., Kaplan, D. and Horton, N. (2014). *"mosaic: Project MOSAIC (mosaic-web.org) statistics and mathematics teaching utilities"*. R package version 0.9.1-3.
URL: *http://CRAN.R-project.org/package=mosaic, last accessed April 11, 2015*

Rudder, C. (2010). "The Biggest Lies in Online Data", *OkTrends: dating research from OkCupid* .
URL: *http://blog.okcupid.com/index.php/the-biggest-lies-in-online-dating, last accessed April 11, 2015*

Rudder, C. (2014). *Dataclysm: Who We Are When We Think No One's Looking*, Crown.

Webb, A. (2013). "How I Hacked Online Dating", *TED Talks* .

Wickham, H. (2009). *ggplot2: elegant graphics for data analysis*, Springer New York.
URL: *http://had.co.nz/ggplot2/book*

Wickham, H. (2012). *"stringr: Make it easier to work with strings"*. R package version 0.6.2.
URL: *http://CRAN.R-project.org/package=stringr, last accessed April 11, 2015*

Wickham, H. (2014). "How are Data Science and Statistics different?", *IMS Bulletin* **43**(6).
URL: *http://bulletin.imstat.org/2014/09/data-science-how-is-it-different-to-statistics/, last accessed April 11, 2015*

Wickham, H. and Francois, R. (2014). *"dplyr: a grammar of data manipulation"*. R package version 0.2.
URL: *http://CRAN.R-project.org/package=dplyr, last accessed April 11, 2015*

Yu, B. (2014). "IMS Presidential Address: Let Us Own Data Science", *IMS Bulletin* **43**(7).
URL: *http://bulletin.imstat.org/2013/10/president%E2%80%99s-welcome-bin-yu/, last accessed April 11, 2015*