Descriptive data tables

Paul Johnson, CRMDA, pauljohn@ku.edu

Guide No: 0 **Keywords**: R **Sept. 11, 2018**
See https://pj.freefaculty.org/guides for updates.

Abstract

This is an abstract. Please include a terse, yet descriptive statement here of less than 200 words. It should avoid colloquialisms and polysyllabic profundities.

Contents

1425 Jayhawk Blvd.
Watson Library, Suite 470
Lawrence, KS 66045-7594

Web: https://crmda.ku.edu
Email: crmda@ku.edu
Phone: 785-864-3353

1 Reviewers want descriptive variable tables

People who read our models usually want to know about the variables that are included in the model. In the very simplest case, it might require only the mean and standard deviation of each predictor. That kind of table is very easy to produce, as we will see.

A more complicated scenario arises when there are different sorts of variables to be summarized and the researcher wants to knit them together.

1.1 The Social Welfare Project

We had a funded project that replicated a previous research project.

In Table 1, please see an example from an article by Nam (2008) that we were required to replicate. From updated data, we generated Table 2. Our output table is reasonable. I checked the R code that was used to generate it and it is very complicated; it is not the sort of simple, elegant code I'd want to teach you to use today. That example is complicated because it is flexible to deal with different yearly data sets and there are several variable types.

Table 1: Summary table on social welfare project

TABLE 2

Descriptive Statistics for the Full Sample, Target Group, and Comparison Group

Nam (2008) SSQ

	Full Sample	Target (Female Heads with Children)	Comparison (Male Heads and Female Heads Without Children)
On welfare in 1994***	0.06	0.39	0.01
On welfare between 1994–2001***	0.08	0.48	0.02
Age***	34.39	31.80	34.76
African-American***	0.20	0.61	0.14
Head's education in 1994***			
Less than high school	0.13	0.29	0.11
High school degree	0.51	0.37	0.53
Some college	0.36	0.34	0.37
Household size in 1994***	2.90	3.21	2.86
Change in household size (1994–2001)	0.01	0.06	0.00
Number of children***	1.19	2.08	1.06
Averaged family income (1994–2001)***			
Mean	$38,709.35	$13,740.67	$42,357.22
Median	$33,841.96	$11,859.56	$39,080.11
Change in family income (1994–2001)			
Mean	$9,184.63	$9,651.91	$9,116.36
Median	$6,287.39	$7,126.97	$6,138.73
Change in state unemployment rates (1994–2001)	−1.20	−1.34	−1.18
Per capita GSP in 1994 (in $1,000)*	27.32	28.03	27.22
Financial assets in 1994**			
Mean	$17,191	$1,998	$19,411
Median	$1,588	$0	$2,117
Change in financial assets (1994–2001)			
Mean	$4,829	$1,433	$5,326
Median	$0	$0	$9
Saved financial assets (1994–2001)*	0.50	0.41	0.51
Possessed bank account in 1994***	0.72	0.38	0.77
Possessed bank account in 2001***	0.78	0.58	0.81
Owned a vehicle in 1994***	0.84	0.62	0.87
Owned a vehicle in 2001***	0.87	0.73	0.89
N	1,363	277	1,086

*$p<0.1$; **$p<0.05$; ***$p<0.01$, t tests and χ^2 tests of differences between the target and comparison groups.

Table 2: Updated social welfare data (project 663)

	Full Sample	Target Group	Comparison Group
On welfare in 2002	0.03	0.19	0.00
On welfare between 2002 and 2012	0.03	0.12	0.00
Age	34.03	31.18	34.61
African American	0.40	0.83	0.31
Head's education in 2003			
–Less than high school	0.14	0.27	0.11
–High school degree	0.44	0.40	0.45
–Some College	0.42	0.33	0.44
Household size in 2003	3.20	3.44	3.15
Change in household size (2003-2013)	-0.02	-0.02	-0.01
Number of children	1.35	2.19	1.18
Average family income (2002-2012)			
–Mean	$54,607.45	$22,265.05	$61,133.81
–Median	$44,449.76	$18,266.60	$52,327.77
Change in family income (2002-2012)			
–Mean	$3,819.98	$2,925.92	$4,000.85
–Median	$3,160.13	$4,474.76	$4,232.61
Change in state unemployment rate (2003-2013)	1.42	1.54	1.40
Per capita GSP in 2003	$43,951.73	$42,827.91	$44,179.07
Financial assets in 2003:			
–Mean	$4,926.50	$879.27	$5,766.16
–Median	$500.00	$0.00	$1,000.00
Change in financial assets (2003-2013)			
–Mean	$1,630.13	$-400.15	$2,029.70
–Median	$-26.08	$0.00	$-52.15
Possessed savings in 2003	0.66	0.34	0.72
Possessed savings in 2013	0.62	0.28	0.69
Owned a vehicle in 2003	0.87	0.65	0.91
Owned a vehicle in 2013	0.86	0.73	0.89
N	1153	194	959

As I look at these examples, I realize there are a couple of very important details.

1. For which rows of data should the summary be calculated?

   As you may know, many stats programs use "listwise deletion" of cases with missing values. That means the rows of data that are used may differ between models.

   Probably, we want the summary to reflect the data that is actually used in a model. Possibly/Probably we also might want a summary of the whole data set.

2. Which summary indicators are required?

   The social welfare project is tricky because they wanted different summary items for different kinds of predictors

3. What are we supposed to do about categorical variables?

2 The Workflow Problem

Clearly, we want the software to write a table on disk in a format that can be inserted into a document without trouble.

There are two scenarios.

1. Three step process
   - Run an R file that writes a file on disk (write into output folder)
   - Copy that file into the writeup folder, *possibly with some hand editing.*
   - Create the writeup document that imports that table.
2. One step process: reproducible research document
   - The code in the document produces tables and figures that *do NOT need to hand editing.* They are ready as is!
   - The documents in the stationery package demonstrate this idea.
   - See stationery vignette "Code Chunks" because it shows how *as is* material can be used in different kinds of document

2.1 Formatting for tables

The desired output format depends on the output document format. If we are creating a PDF document, best to have:

LaTeX formatting

If we are creating a Web Page, then best to have:

HTML formatting

If we are completely lost, we may have to take the worst possible avenue:

raw "CSV" formatting

This last is the worst option because the output is not structured and cannot be used in a document without a lot of hand editing.

This document is a LaTeX/NoWeb document that has the PDF back end, so I'm saving table files as LaTeX tabular objects.

Authors have some control over the features and formatting that will be inserted into the table. In my documents, I prefer to create the floating table and figure objects and then insert the results in them. Hence, I do not want my LaTeX table output file to include caption or label. That is a matter of taste, sometimes I will write the table (or figure) captions into the files.

The LaTeX table writing function I prefer is a traditional one, `xtable`. It can create either LaTeX or HTML output. It has worked well for many years, I (honestly) don't understand why so many R package writers want to re-invent this ability.

If I'm writing a regression table, I generally use my function, `outreg` in the `rockchalk` package. However, that works for only the standard regression models provided with R. Some user-contributed regression packages do not create the required structures from which `outreg` will work.

3 Implementation: Numeric Predictors

Lets test with R (R Core Team, 2018). There are many data summary tools in R. I'm partial to the function "summarize" in the rockchalk package, but I expect you can find many packages that get the same work done.

In short, 3 steps are needed.

1. Tabulate data summaries. The will often be in R data.frames, matrices, or similar. Sometimes there will be a list of summaries by subgroups.
2. Inspect the data in the R session, make sure it looks correct.
3. Use one of the R functions that can write the file in the needed format.

3.1 First, check what the rockchalk::summarize function is doing

Read ?summarize, it explains the output printed on screen is just the pretty rendition, but there is actually a list provided as the return. Within the list, the first 2 items are a numeric summary and a summary of the categorical variables.

3.2 Mean and Standard Deviations

We hope guide authors will choose carefully thought out titles for sections and that material will be grouped meaningfully into sections.

```
odir <- "output"
if(!file.exists(odir)) dir.create(odir)
wd <- "workingdata"
fn <- "hsb2.rds"
hsb <- readRDS(file.path(wd, fn))
```

The output from summarize in R is wide, does not fit into this document unless I make the font small.

```
library(rockchalk)
sum.hsb <- summarize(hsb)
```

```
Numeric variables
             ses     mathach      size    pracad    disclim   himinty   schoolid      mean
min          -3.76     -2.83    100         0        -2.42       0       1224          4.24
med           0        13.13   1016         0.53     -0.23       0       5192         13.16
max           2.69     24.99   2713         1         2.76       1       9586         19.72
mean          0        12.75   1056.86      0.53     -0.13       0.28    5277.90      12.75
sd            0.78      6.88    604.17      0.25      0.94       0.45    2499.58       3.01
skewness     -0.23     -0.18      0.57      0.16      0.24       0.98       0.11      -0.27
kurtosis     -0.38     -0.92     -0.36     -0.89     -0.16      -1.04      -1.25      -0.05
nobs       7185      7185      7185      7185      7185       7185       7185       7185
nmissing      0         0         0         0         0          0          0          0
             sd       sdalt      junk    sdalt2      num        se        sealt    sealt2
min           3.54      6.26      0        48.39     14          0.51       0.76      0.85
med           6.30      6.26     30.63     48.39     51          0.89       0.88      0.97
max           8.48      6.26    239.29     48.39     67          1.82       1.67      1.86
```

```
mean          6.20      6.26     47.32     48.39     48.02      0.92      0.92      1.03
sd            0.86      0        48.90      0        10.82      0.20      0.13      0.14
skewness     -0.24       NaN      1.30       NaN     -0.58      1.11      1.59      1.59
kurtosis      0.22       NaN      1.46       NaN     -0.37      2.32      3.25      3.25
nobs       7185      7185      7185      7185      7185      7185      7185      7185
nmissing      0         0         0         0         0         0         0         0
             t2       t2alt    pickone    mmses     mnses       xb       resid
min           0         0         0        -1.19     -1.19      5.68    -19.49
med           5.75      4.36      0         0.03      0.03     12.87      0.24
max         195.81     52.82      1         0.82      0.82     17.52     16.44
mean         14.66      8.54      0.02      0         0        12.69      0.06
sd           26.42     11.06      0.15      0.41      0.41      2.42      6.46
skewness      3.67      2.06      6.47     -0.27     -0.27     -0.27     -0.14
kurtosis     16.89      4.24     39.92     -0.48     -0.48     -0.48     -0.69
nobs       7185      7185      7185      7185      7185      7185      7185
nmissing      0         0         0         0         0         0         0

Nonnumeric variables
      sector          gender            ethnicity              schoolidf
 Public  : 3642   Female: 3795     White    : 5211     2305        :   67
 Catholic: 3543   Male  : 3390     Non-white: 1974     5619        :   66
 nobs    : 7185   nobs  : 7185     nobs     : 7185     4292        :   65
 nmiss   :    0   nmiss :    0     nmiss    :    0     3610        :   64
 entropy      : 1 entropy      : 1 entropy      : 0.85 (All Others): 6923
 normedEntropy: 1 normedEntropy: 1 normedEntropy: 0.85 nobs        : 7185
                                                       nmiss       :    0
                                                       entropy      : 7.27
                                                       normedEntropy: 0.99
```

The output object sum.hsb has 3 parts, the third one is that thing you are looking at. It is the formatted, "textified" version. The first 2 parts are the "numeric" and "categorical" predictors, which are not rounded or summarized.

Take a look. My numeric summary has the variables on the rows, not on the columns as shown in the printed output within the R session:

```
sum.hsb[[1]]
```

```
                  min          med          max          mean           sd   skewness
ses      -3.758000e+00    0.0020000    2.6919999  1.433540e-04    0.7793552 -0.2280971
mathach  -2.832000e+00   13.1309996   24.9930000  1.274785e+01    6.8782457 -0.1804918
size      1.000000e+02 1016.0000000 2713.0000000  1.056862e+03  604.1724993  0.5714961
pracad    0.000000e+00    0.5300000    1.0000000  5.344871e-01    0.2511861  0.1595830
disclim  -2.416000e+00   -0.2310000    2.7560000 -1.318694e-01    0.9439882  0.2394417
himinty   0.000000e+00    0.0000000    1.0000000  2.800278e-01    0.4490438  0.9795996
schoolid  1.224000e+03 5192.0000000 9586.0000000  5.277898e+03 2499.5777954  0.1073614
mean      4.239781e+00   13.1601057   19.7191429  1.274785e+01    3.0058166 -0.2711762
sd        3.541020e+00    6.2984834    8.4811230  6.197527e+00    0.8637071 -0.2357613
sdalt     6.256328e+00    6.2563276    6.2563276  6.256328e+00    0.0000000        NaN
junk      2.473890e-05   30.6254406  239.2891541  4.731597e+01   48.8976099  1.2998278
sdalt2    4.839363e+01   48.3936348   48.3936348  4.839363e+01    0.0000000        NaN
num       1.400000e+01   51.0000000   67.0000000  4.801628e+01   10.8221802 -0.5792808
se        5.058600e-01    0.8938972    1.8237413  9.189899e-01    0.2017056  1.1131857
sealt     7.643321e-01    0.8760611    1.6720738  9.246300e-01    0.1291964  1.5868761
sealt2    8.498783e-01    0.9741123    1.8592170  1.028117e+00    0.1436564  1.5868762
t2        7.423064e-04    5.7493305  195.8105927  1.465580e+01   26.4160072  3.6683986
t2alt     7.485392e-04    4.3591580   52.8245697  8.537593e+00   11.0627385  2.0632127
pickone   0.000000e+00    0.0000000    1.0000000  2.226862e-02    0.1475661  6.4739108
mmses    -1.193946e+00    0.0320000    0.8249825  1.433532e-04    0.4135432 -0.2684650
mnses    -1.193946e+00    0.0320000    0.8249825  1.433532e-04    0.4135432 -0.2684650
xb        5.683861e+00   12.8722429   17.5219269  1.268545e+01    2.4248264 -0.2684650
resid    -1.948889e+01    0.2357960   16.4445744  6.240260e-02    6.4594589 -0.1359769
            kurtosis nobs nmissing
ses      -0.38044986 7185        0
mathach  -0.92159871 7185        0
```

```
size      -0.36494527 7185        0
pracad    -0.88590959 7185        0
disclim   -0.15882341 7185        0
himinty   -1.04052940 7185        0
schoolid -1.25461841 7185        0
mean      -0.05066915 7185        0
sd         0.21576353 7185        0
sdalt             NaN 7185        0
junk       1.46197638 7185        0
sdalt2            NaN 7185        0
num       -0.37038907 7185        0
se         2.32256458 7185        0
sealt      3.24962919 7185        0
sealt2     3.24962964 7185        0
t2        16.88833532 7185        0
t2alt      4.24329712 7185        0
pickone   39.91707701 7185        0
mmses     -0.47890987 7185        0
mnses     -0.47890987 7185        0
xb        -0.47890979 7185        0
resid     -0.68558788 7185        0
```

The easy thing is to summarize the numeric variables. In the output folder, it will write "hsbsumry20.tex".

```
sum.hsb.2 <- sum.hsb[[1]][c("mean", "sd")]
library(xtable)
xt <- xtable(sum.hsb.2)
```

```
print(xt, file = file.path(odir, "hsbsumry20.tex"),
       floating=FALSE)
```

I'm going to create a "floating" table, Table 3.

Table 3: HSB summary information

	mean	sd
ses	0.00	0.78
mathach	12.75	6.88
size	1056.86	604.17
pracad	0.53	0.25
disclim	-0.13	0.94
himinty	0.28	0.45
schoolid	5277.90	2499.58
mean	12.75	3.01
sd	6.20	0.86
sdalt	6.26	0.00
junk	47.32	48.90
sdalt2	48.39	0.00
num	48.02	10.82
se	0.92	0.20
sealt	0.92	0.13
sealt2	1.03	0.14
t2	14.66	26.42
t2alt	8.54	11.06
pickone	0.02	0.15
mmses	0.00	0.41
mnses	0.00	0.41
xb	12.69	2.42
resid	0.06	6.46

4 Categorical Variables: more difficult

4.1 Likert variables: I have that worked out!

In the `kutils` package, I wrote a function called likert that can line up responses to many Likert scale questions and then assemble them into a table. The output here is actually showing 2 tables, because there are too many columns to fit in 1 table.

Table 4: Likert variable summary from TFA project

Table 3: NCS Item Frequencies

	NCS1	NCS2	NCS3	NCS4
1	0.30% (2)	1.80% (12)	0.75% (5)	0.60% (4)
2	0.60% (4)	5.69% (38)	3.44% (23)	2.69% (18)
3	2.99% (20)	11.83% (79)	5.69% (38)	5.99% (40)
4	4.79% (32)	6.59% (44)	7.49% (50)	8.68% (58)
5	18.26% (122)	24.10% (161)	17.81% (119)	20.21% (135)
6	57.04% (381)	36.23% (242)	45.06% (301)	39.82% (266)
7	16.02% (107)	13.77% (92)	19.76% (132)	22.01% (147)
Total	668	668	668	668
	NCS7	NCS8	NCS9	NCS10
1	2.25% (15)	1.35% (9)	4.04% (27)	0.75% (5)
2	4.79% (32)	3.44% (23)	8.68% (58)	4.64% (31)
3	7.04% (47)	7.78% (52)	10.03% (67)	5.69% (38)
4	7.19% (48)	8.98% (60)	10.18% (68)	9.43% (63)
5	24.10% (161)	20.66% (138)	22.16% (148)	25.45% (170)
6	38.32% (256)	34.73% (232)	27.40% (183)	37.43% (250)
7	16.32% (109)	23.05% (154)	17.51% (117)	16.62% (111)
Total	668	668	668	668

4.2 More generally, more difficult

The output from rockchalk::summarize has a separate piece for the categorical variables. It is a list, with one table for each variable.

```
sum.hsb[[2]]
```

```
     sector          gender             ethnicity              schoolidf
Public  : 3642   Female: 3795     White     : 5211     2305         :   67
Catholic: 3543   Male  : 3390     Non-white: 1974     5619         :   66
nobs    : 7185   nobs  : 7185     nobs      : 7185     4292         :   65
nmiss   :    0   nmiss :    0     nmiss     :    0     3610         :   64
entropy      : 1 entropy      : 1 entropy       : 0.85 (All Others): 6923
normedEntropy: 1 normedEntropy: 1 normedEntropy: 0.85 nobs         : 7185
                                                      nmiss        :    0
                                                      entropy      : 7.27
                                                      normedEntropy: 0.99
```

I notice that the output from this command is not handled well by the LaTeX to PDF transition, so I won't run it.

```
str(sum.hsb[[2]])
```

If we want to summarize the different tables, we have to make some document preparation decisions.

- Which summary information do we require?

- Are we trying to squash all of the tables into one big summary table?

The way I created those summary tables causes complications throughout. I need to think about how to fix.

4.3 Pretty easy to make separate summaries

Here, we will be like people who lived in caves and write summary code for each of the first 3 table objects. I am NOT writing those summaries in separate files. I am using the LaTeX results "as is", they are printing directly into the document.

```
library(xtable)
xt1 <- xtable(as.table(sum.hsb[[2]][["sector"]]$table),
              caption = "The Sector Summary", label =
                 "tab:sector",
              digits = 2)
print(xt1, table.placement="H", caption.placement="top")
```

Table 5: The Sector Summary

	x
Public	3642
Catholic	3543
nobs	7185
nmiss	0

```
xt2 <- xtable(as.table(sum.hsb[[2]][["gender"]]$table),
              caption = "The Gender Summary", label =
                 "tab:gender",
              digits = 2)
print(xt2, table.placement="H", caption.placement="top")
```

Table 6: The Gender Summary

	x
Female	3795
Male	3390
nobs	7185
nmiss	0

```
xt3 <- xtable(as.table(sum.hsb[[2]][["ethnicity"]]$table),
              caption = "The Ethnicity Summary", label =
                 "tab:ethnicity",
              digits = 2)
print(xt3, table.placement="H", caption.placement="top")
```

Table 7: The Ethnicity Summary

	x
White	5211
Non-white	1974
nobs	7185
nmiss	0

5 Get out of Jail Free Card? Regression Model Matrix?

In R, a regression model will, by default, have the abilty to give back a model design matrix. The function for this is `model.matrix`.

Suppose we fit a multiple regression, and then use model.matrix to recover the data and the recoded variables that were actually used.

I'm not claiming this is a reasonable regression, but it will run and demonstrate the purpose.

```
m1 <- lm(mathach ~ ses + size + sector + gender + ethnicity, data
   = hsb)
```

The raw output of the summary.lm method function is like this:

```
summary(m1)
```

```
Call:
lm(formula = mathach ~ ses + size + sector + gender + ethnicity,
    data = hsb)

Residuals:
     Min       1Q   Median       3Q      Max
-20.1204  -4.4920   0.2334   4.7397  17.3453

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        12.3472389  0.2190316  56.372  < 2e-16 ***
ses                 2.3403235  0.0993873  23.548  < 2e-16 ***
size                0.0006893  0.0001338   5.153 2.63e-07 ***
sectorCatholic      2.6193556  0.1647511  15.899  < 2e-16 ***
genderFemale       -1.3928702  0.1459261  -9.545  < 2e-16 ***
ethnicityNon-white -3.2183223  0.1712270 -18.796  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.155 on 7179 degrees of freedom
Multiple R-squared:  0.1999,  Adjusted R-squared:  0.1993
F-statistic: 358.7 on 5 and 7179 DF,  p-value: < 2.2e-16
```

In Table 8, I have a nicer-looking summary.

Table 8: Outreg from rockchalk package

```
outreg(list("One regression I ran" = m1), tight = FALSE)
```

	One regression I ran	
	Estimate	(S.E.)
(Intercept)	12.347***	(0.219)
ses	2.340***	(0.099)
size	0.001***	(0.000)
sectorCatholic	2.619***	(0.165)
genderFemale	-1.393***	(0.146)
ethnicityNon-white	-3.218***	(0.171)
N	7185	
RMSE	6.155	
R^2	0.200	
adj R^2	0.199	

$*p \leq 0.05** p \leq 0.01***p \leq 0.001$

```
hsb.mm1 <- model.matrix(m1)
```

Note that all of the columns in the `model.matrix` output are numeric. Even the categorical predictors are reduced to 1's and 0's.

```
head(hsb.mm1)
```

```
  (Intercept)    ses size sectorCatholic genderFemale ethnicityNon-white
1           1 -1.528  842              0            1                   0
2           1 -0.588  842              0            1                   0
3           1 -0.528  842              0            0                   0
4           1 -0.668  842              0            0                   0
5           1 -0.158  842              0            0                   0
6           1  0.022  842              0            0                   0
```

Why not grab the numerical summaries of all of those predictors as seen from the `model.matrix` point of view?

```
## convert from matrix to data frame needed for summarize
hsb.mm1 <- as.data.frame(hsb.mm1)
hsb.smry2 <- rockchalk::summarize(hsb.mm1)
```

```
Numeric variables
          (Intercept)     ses       size    sectorCatholic   genderFemale
min             1          -3.76    100          0                0
med             1           0      1016          0                1
max             1           2.69   2713          1                1
mean            1           0      1056.86       0.49             0.53
sd              0           0.78    604.17       0.50             0.50
skewness         NaN       -0.23      0.57       0.03            -0.11
kurtosis         NaN       -0.38     -0.36      -2               -1.99
nobs         7185          7185      7185       7185             7185
nmissing        0             0         0          0                0
          ethnicityNon-white
min             0
med             0
max             1
```

```
mean          0.27
sd            0.45
skewness      1.01
kurtosis     -0.98
nobs       7185
nmissing      0
```

Note there are no factors there. Lets see if output in Table **??** is close to good.

```
## I just want mean and sd, so pick those out
hsb.smry3 <- hsb.smry2[[1]][ , c("mean", "sd")]
## I dial up digits here
xt3 <- xtable(hsb.smry3, caption="xtable output from model.matrix
   output",
        label = "tab:sumry3", digits=4)
options(scipen=20) ## I don't want scientific notation
print(xt3, table.placement="H", caption.placement="top")
```

Table 9: xtable output from model.matrix output

	mean	sd
(Intercept)	1.0000	0.0000
ses	0.0001	0.7794
size	1056.8618	604.1725
sectorCatholic	0.4931	0.5000
genderFemale	0.5282	0.4992
ethnicityNon-white	0.2747	0.4464

If that's close to good, I suggest we write it in a file, and then hand edit the variable names, then include it in a document:

```
print(xt3, digits = 2, table.placement="H",
   caption.placement="top",
       file = file.path(odir, "mm.sumry3.tex"))
```

Manually copy that into some other folder where we can cultivate and perfect it. I often use a folder named "importfigs" for that kind of thing.

The output in Table 9 is almost good enough. Each of the "dummy variables" in the regression has a mean equal the proportion of 1's in that predictor column. If we had a multi-category predictor, then we'd see more than one row for each variable. Still, there's a problem that it "leaves out" one category because it is "in the intercept." We can think on that.

Actually, I thought about that, here's my fix in Table 10. If that has all of the right numbers, then we should write it in a file, fix the labels, and be done with it.

```
m1.noint <- lm(mathach ~ -1 +  ses + size + sector + gender +
   ethnicity, data = hsb)
hsb.mm1.noint <- model.matrix(m1.noint)
hsb.mm1.noint <- as.data.frame(hsb.mm1.noint)
hsb.smry.noint <- rockchalk::summarize(hsb.mm1.noint)
```

```
Numeric variables
             ses       size    sectorPublic   sectorCatholic   genderFemale
min          -3.76    100         0              0                0
med           0      1016         1              0                1
max           2.69   2713         1              1                1
mean          0      1056.86      0.51           0.49             0.53
sd            0.78    604.17      0.50           0.50             0.50
skewness     -0.23      0.57     -0.03           0.03            -0.11
kurtosis     -0.38     -0.36     -2             -2               -1.99
nobs       7185      7185      7185           7185             7185
nmissing      0         0         0              0                0
           ethnicityNon-white
min           0
med           0
max           1
mean          0.27
sd            0.45
skewness      1.01
kurtosis     -0.98
nobs       7185
nmissing      0
```

```
hsb.smry.noint <- hsb.smry.noint[[1]][ , c("mean", "sd")]
```

```
xt.noint <- xtable(hsb.smry.noint, caption="Summary showing all
   categories",
        label = "tab:sumry.noint", digits=4)
options(scipen=20) ## I don't want scientific notation
print(xt.noint, table.placement="H", caption.placement="top")
```

Table 10: Summary showing all categories

	mean	sd
ses	0.0001	0.7794
size	1056.8618	604.1725
sectorPublic	0.5069	0.5000
sectorCatholic	0.4931	0.5000
genderFemale	0.5282	0.4992
ethnicityNon-white	0.2747	0.4464

5.1 Create 1 function to do approximately the correct thing

```
dat <- genCorrelatedData2(1000, means=c(10, 10, 10), sds = 3, stde
   = 3, beta = c(1, 1, -1, 0.5))
```

```
[1] "The equation that was calculated was"
y = 1 + 1*x1 + -1*x2 + 0.5*x3
 + 0*x1*x1 + 0*x2*x1 + 0*x3*x1
 + 0*x1*x2 + 0*x2*x2 + 0*x3*x2
 + 0*x1*x3 + 0*x2*x3 + 0*x3*x3
 + N(0,3) random error
```

```
dat$xcat1 <- factor(sample(c("a", "b", "c", "d"), 1000,
   replace=TRUE))
```

```
dat$xcat2 <- factor(sample(c("M", "F"), 1000, replace=TRUE),
   levels = c("M", "F"), labels = "Male", "Female")
dat$y <- dat$y + contrasts(dat$xcat1)[dat$xcat1, ] %*% c(0.1, 0.2,
   0.3)
m4 <- lm(y ~ x1 + x2  + x3 + xcat1 + xcat2, dat)
```

```
##' Summary stats table-maker for regression users
##'
##' rockchalk::summarize does the numerical calculations
##'
##' This is, roughly speaking, doing the right thing, but
##' not in a clever way. I need to think harder on that.
##'
##' @param object A fitted regression or an R data.frame, or any
##'      other object type that does not fail in
   code{model.frame(object)}.
##' @param stats Default is a vector c("mean", "sd", "min",
   "max"). Other
##'      stats reported by rockchalk::summarize should work fine
   as well
##' @param digits 2 decimal points is default
##' @param ... Other arguments passed to
   rockchalk::summarizeNumerics and
##'      summarizeFactors.
##' @return a character matrix
##' @author Paul Johnson
descriptiveTable <- function(object, stats = c("mean", "sd",
   "min", "max"),
                             digits = 2, probs = c(0, .5, 1), ...){
     mc <- match.call(expand.dots = TRUE)
     dots <- list(...)

     dat <- model.frame(object)
     arglist <- list(dat = dat, stats = stats, digits = digits)
     arglist <- modifyList(arglist, dots)
     summ.dat <- do.call(rockchalk::summarize, arglist)
     reslt <- data.frame(variable =
        rownames(summ.dat[["numerics"]]),
                                    summ.dat[["numerics"]][stats[stats
                                       %in%
                                       names(summ.dat[["numerics"]])]],
                         stringsAsFactors = FALSE)
     numbers <- names(which(sapply(reslt, is.numeric)))
     for(j in numbers) reslt[ , j] <- formatC(reslt[ , j], digits
        = digits)

     reslt2 <- vector("list", length =
        length(summ.dat[["factors"]]))
```

```
    names(reslt2) <- names(summ.dat[["factors"]])
    for(j in names(summ.dat[[2]])){
        tab <- summ.dat[[2]][[j]]
        tab.prop <- tab[["table"]]/tab[["table"]]["nobs"]
        ## remove elements after nobs
        nobs.col <- which(names(tab.prop) == "nobs")
        tab.prop <- tab.prop[1:(nobs.col - 1)]
        reslt2[[j]] <- data.frame(variable=names(tab.prop),
                                  mean = formatC(tab.prop, digits
                                     = digits),
                                  stringsAsFactors = FALSE)
        reslt2[[j]] <- rbind(data.frame(variable = j, mean = "",
                                        stringsAsFactors = FALSE),
                             reslt2[[j]])
    }

    reslt3 <- do.call(rbind, reslt2)

    reslt4 <- plyr::rbind.fill(reslt, reslt3)
    reslt4[is.na(reslt4)] <- ""
    reslt4
}
```

```
m4.desc <- descriptiveTable(m4)
```

```
library(xtable)
m4.desc.tab <- xtable(m4.desc, caption="Testing descriptiveTable",
   label = "tab:makedesc100")
## Put one copy in a file
print(m4.desc.tab, file = file.path(odir, "makedesc100.tex"),
   include.rownames = FALSE,
       table.placement="H", caption.placement="top")
## print another copy to screen
print(m4.desc.tab, include.rownames = FALSE,
       table.placement="H", caption.placement="top")
```

Table 11: Testing descriptiveTable

variable	mean	sd	min	max
y	6.1	5.4	-11	22
x1	9.9	3	-0.004	19
x2	10	3	0.57	20
x3	10	2.9	0.8	20
xcat1				
b	0.27			
a	0.26			
c	0.25			
d	0.23			
xcat2				
Male2	0.53			
Male1	0.47			

```
m4.desc2 <- descriptiveTable(m4, stats = c("mean", "var",
   "skewness"), probs = c(0.0, 0.2, 0.8, 1))
```

Check contents of m4.desc2

```
m4.desc2
```

```
   variable mean var skewness
1         y  6.1  29   -0.029
2        x1  9.9 8.7    -0.11
3        x2   10 9.2     0.027
4        x3   10 8.5     -0.06
5     xcat1
6         b 0.27
7         a 0.26
8         c 0.25
9         d 0.23
10    xcat2
11    Male2 0.53
12    Male1 0.47
```

```
m4.desc.tab2 <- xtable(m4.desc2, caption="Testing
   descriptiveTable", label = "tab:makedesc100")
print(m4.desc.tab2, include.rownames = FALSE,
       table.placement="H", caption.placement="top")
```

Table 12: Testing descriptiveTable

variable	mean	var	skewness
y	6.1	29	-0.029
x1	9.9	8.7	-0.11
x2	10	9.2	0.027
x3	10	8.5	-0.06
xcat1			
b	0.27		
a	0.26		
c	0.25		
d	0.23		
xcat2			
Male2	0.53		
Male1	0.47		

Many problems still remain and while fixing them I have to revise the rockchalk::summarize function itself. Thus, when that is finished, I'll have to come back here and make descriptiveTable compatible again.

So check back soon, as I will fix this up to be Great Again, just like America.

References

R Core Team (2018). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria: R Foundation for Statistical Computing.

Replication Information

Please leave this next code chunk if you are producing a guide document.

```
R version 3.5.1 (2018-07-02)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 18.04.1 LTS

Matrix products: default
BLAS: /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.7.1
LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.7.1

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C               LC_TIME=en_US.UTF-8
 [4] LC_COLLATE=en_US.UTF-8     LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C                  LC_ADDRESS=C
[10] LC_TELEPHONE=C             LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] xtable_1.8-2        rockchalk_1.8.115   stationery_0.98.5.4

loaded via a namespace (and not attached):
```

```
 [1] zip_1.0.0          Rcpp_0.12.17       nloptr_1.0.4       compiler_3.5.1
 [5] pillar_1.2.3       cellranger_1.1.0   plyr_1.8.4         forcats_0.3.0
 [9] tools_3.5.1        lme4_1.1-17        digest_0.6.15      nlme_3.1-137
[13] lattice_0.20-35    evaluate_0.10.1    tibble_1.4.2       rlang_0.2.1
[17] Matrix_1.2-14      openxlsx_4.1.0     curl_3.2           pbivnorm_0.6.0
[21] haven_1.1.1        rio_0.5.10         stringr_1.3.1      knitr_1.20
[25] grid_3.5.1         stats4_3.5.1       rprojroot_1.3-2    data.table_1.11.4
[29] readxl_1.1.0       foreign_0.8-70     rmarkdown_1.10     lavaan_0.6-1
[33] minqa_1.2.4        carData_3.0-1      car_3.0-0          magrittr_1.5
[37] splines_3.5.1      backports_1.1.2    htmltools_0.3.6    MASS_7.3-50
[41] kutils_1.46        abind_1.4-5        mnormt_1.5-5       stringi_1.2.3
```