Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 9
dat <- read.dta(paste("../student-test2/student-",i,".dta", sep = ""))
```

The variables pprof and pnet are scored as numeric, but really they are factors. So convert them to prevent future mis-understandings.

```
dat$pprof <- factor(dat$pprof, labels = c("NO", "YES"))
dat$pnet <- factor(dat$pnet, labels = c("NO","YES"))
```

```
datsum <- summarize(dat)
```

Table would need some hand customization

```
library(xtable)
print(xtable(datsum$numeric, caption = "Best Automatic Summary Table for Numerics", label =
    "table1"), "latex")
```

	act	harv	ibs	sal1	sal2	sal3	sat
0%	9.53	908.80	72.36	3026.00	3033.00	150200.00	901.90
25%	18.70	1507.00	93.15	16290.00	19000.00	161300.00	1485.00
50%	21.90	1620.00	99.73	20090.00	23300.00	165300.00	1603.00
75%	25.02	1722.00	106.50	23670.00	26820.00	169800.00	1702.00
100%	35.91	2146.00	129.50	33990.00	39970.00	183700.00	2124.00
mean	21.90	1614.00	99.66	20130.00	23050.00	165500.00	1597.00
sd	4.90	161.30	9.60	5307.00	5595.00	6101.00	161.40
var	24.01	26030.00	92.14	28160000.00	31310000.00	37220000.00	26030.00
NA's	15.00	52.00	0.00	9.00	0.00	0.00	33.00
N	573.00	573.00	573.00	573.00	573.00	573.00	573.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

```
print(datsum$factors)
```

```
          gender                   major                    pnet
F             :298.0000   H             :204.0000   NO            :392.0000
M             :275.0000   N             :192.0000   YES           :181.0000
NA's          :  0.0000   S             :177.0000   NA's          :  0.0000
entropy       :  0.9988   NA's          :  0.0000   entropy       :  0.8998
normedEntropy:  0.9988   entropy       :  1.5825   normedEntropy:  0.8998
N             :573.0000   normedEntropy:  0.9985   N             :573.0000
                          N             :573.0000
          pprof
NO            :394.0000
YES           :179.0000
NA's          :  0.0000
entropy       :  0.8959
normedEntropy:  0.8959
N             :573.0000
```

Aptitude Test Variables

There's severe multicollinearity between the variables harv, sat, and act. It seems clear we can't estimate both sat and harv, and several students noticed that since harv is a summary of the other tests, then there's some reason to suppose sat is a better variable. (I know for a fact that harv = sat + act).

Please find Table 2. I left the Iowa Basic Skills variable in my best model, mainly because I wanted to estimate that coefficient, even though the F test below indicates one can exclude harv and ibs from the "full" model without losing any sleep.

```
m1s <- lm(sal1 ~ sat, data = dat)
m1a <- lm(sal1 ~ act, data = dat)
m1i <- lm(sal1 ~ ibs, data = dat)
m1h <- lm(sal1 ~ harv, data = dat)
m1all <- lm(sal1 ~ sat + act + ibs + harv, data = dat)
m1best <- lm(sal1 ~ sat + act + ibs, data = dat)
```

```
mcDiagnose(m1all)
```

```
The following auxiliary models are being estimated and returned in a list:
sat ~ act + ibs + harv
<environment: 0x1c2a200>
act ~ sat + ibs + harv
<environment: 0x1c2a200>
ibs ~ sat + act + harv
<environment: 0x1c2a200>
harv ~ sat + act + ibs
<environment: 0x1c2a200>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998405 0.8668013 0.2004331 0.9998444
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
6270.118905    7.507581    1.250677 6428.682405
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.40 0.38 1.00
act  0.40 1.00 0.35 0.43
ibs  0.38 0.35 1.00 0.39
harv 1.00 0.43 0.39 1.00
```

```
niceLabels <- c(act = "ACT", sat = "SAT", harv = "Harvard SS", ibs = "Iowa BS",  majorS = "
    Major: Soc.", majorN = "Major: Nat.", majorH = "Major: Hum.", pnetYES = "Parent Network
    : Yes",  pprofYES="Prof. Parents: Yes", genderM = "Gender: Male", "log(harv)"= "ln(
    Harvard SS)",
    "I(harv * harv)"= "Harvard SS$^2$", major2H = "Major 2: Hum.", major2N = "Major 2: Nat.
        ")
outreg(list(m1s, m1a, m1i, m1h, m1all, m1best), tight = TRUE, modelLabels = c("SAT","ACT","
    IBS","Harvard SS", "All", "Best"), varLabels = niceLabels, title = paste("Regression
    with sal1: Student-",i, sep=""), label = "tab:tab2")
```

Could conduct an F test of the hypothesis that $b_{ibs} = b_{harv} = 0$. But which model should I be testing? Test the one with all the variables, to see if *harv* and *ibs* should both be set to 0. To do that, I need to take the data frame used to fit m1all and use it to fit the restricted model. Otherwise, the F test fails.

```
m1alldf <- model.frame(m1all)
m1restricted <- lm(sal1 ~ sat + act, data = m1alldf)
anova(m1restricted, m1all)
```

```
Analysis of Variance Table

Model 1: sal1 ~ sat + act
Model 2: sal1 ~ sat + act + ibs + harv
```

Table 2: Regression with sal1: Student-9

	SAT Estimate (S.E.)	ACT Estimate (S.E.)	IBS Estimate (S.E.)	Harvard SS Estimate (S.E.)	All Estimate (S.E.)	Best Estimate (S.E.)
(Intercept)	2417.694	12714.211*	10736.51*	2475.678	2058.067	2855.441
	(2142.169)	(982.351)	(2307.311)	(2222.098)	(2839.03)	(2627.22)
SAT	11.108*	.	.	.	201.709	8.415*
	(1.335)				(112.613)	(1.533)
ACT	.	339.669*	.	.	406.197*	230.097*
		(43.791)			(125.898)	(48.955)
Iowa BS	.	.	94.221*	.	-11.799	-11.835
			(23.036)		(26.999)	(25.528)
Harvard SS	.	.	.	10.979*	-192.496	.
				(1.37)	(112.542)	
N	531	549	564	513	473	517
RMSE	4984.28	5042.109	5233.92	5024.727	4934.131	4888.353
R^2	0.116	0.099	0.029	0.112	0.155	0.152
adj R^2	0.114	0.097	0.027	0.11	0.148	0.147

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    470 1.1467e+10
2    468 1.1394e+10  2  73028065 1.4998 0.2242
```

Noticing this sample size problem, I wondered if I should re-do Table 2 so that all are fitted on the exact same data. Since I exclude harv, should those cases that are missing on harv "come back to life" when I exclude harv from the model? I think so. Still, there is something unappetizing about this. Fitting harv causes a loss of cases, no matter how we look at it. So for the best model and the ones for sat and ibs, I use the sample from the best model, but when harv enters the picture, we lose some cases.

```
m1best <- lm(sal1 ~ sat + act + ibs, data = dat)
dat2 <- model.frame(m1best)
m1s <- lm(sal1 ~ sat, data = dat2)
m1a <- lm(sal1 ~ act, data = dat2)
m1i <- lm(sal1 ~ ibs, data = dat2)
m1h <- lm(sal1 ~ harv, data = dat[row.names(dat2), ])
m1all <- lm(sal1 ~ sat + act + ibs + harv, data = dat[row.names(dat2), ])
```

```
outreg(list(m1s, m1a, m1i, m1h, m1all, m1best), tight = TRUE, modelLabels = c("SAT","ACT","
    IBS","Harvard SS", "All", "Best"), varLabels = niceLabels)
```

	SAT Estimate (S.E.)	ACT Estimate (S.E.)	IBS Estimate (S.E.)	Harvard SS Estimate (S.E.)	All Estimate (S.E.)	Best Estimate (S.E.)
(Intercept)	2538.424	12795.501*	11124.365*	1623.923	2058.067	2855.441
	(2169.292)	(1004.592)	(2428.475)	(2318.63)	(2839.03)	(2627.22)
SAT	11.031*	.	.	.	201.709	8.415*
	(1.35)				(112.613)	(1.533)
ACT	.	336.302*	.	.	406.197*	230.097*
		(44.694)			(125.898)	(48.955)
Iowa BS	.	.	90.584*	.	-11.799	-11.835
			(24.212)		(26.999)	(25.528)
Harvard SS	.	.	.	11.539*	-192.496	.
				(1.426)	(112.542)	
N	517	517	517	473	473	517
RMSE	4984.45	5028.367	5227.015	5014.32	4934.131	4888.353
R^2	0.115	0.099	0.026	0.122	0.155	0.152
adj R^2	0.113	0.097	0.025	0.12	0.148	0.147

$*p \leq 0.05$

Deciding what's "important"? We have lots of ways. If I've settled on a "best" model, it seems like I should be confined to the variables in that model. And the diagnostics should not depend on harv. Here are the partial and semi-partial correlations.

```
getPartialCor(m1best)
```

```
             sal1
sal1 -1.00000000
sat   0.23559753
act   0.20318893
ibs  -0.02046518
```

```
getDeltaRsquare(m1best)
```

```
The deltaR-square values: the change in the R-square
      observed when a single term is removed.
Same as the square of the 'semi-partial correlation coefficient'
    deltaRsquare
sat 0.0498452565
act 0.0365251843
ibs 0.0003553813
```

I admit, it is tough to conceptualize the scales of the different variables. I suppose I could scale the sat, act, and ibs scores so that they are all on the same 0-100 scale. Then I'll re-run the model. (This is called "percent of maximum" scoring (POMS)). Since we KNOW from previous work that re-scaling a variable has absolutely no substantive impact on the fit, and it is just for convenience of interpretation, this is an innocuous change.

```
dat2$satpoms <- 100*(dat2$sat - min(dat2$sat))/(max(dat2$sat) - min(dat2$sat))
dat2$actpoms <- 100*(dat2$act - min(dat2$act))/(max(dat2$act) - min(dat2$act))
dat2$ibspoms <- 100*(dat2$ibs - min(dat2$ibs))/(max(dat2$ibs) - min(dat2$ibs))
summarize(dat2[, c("satpoms","actpoms","ibspoms")])
```

```
$numerics
     actpoms ibspoms satpoms
0%      0.00    0.00    0.00
25%    34.84   36.88   47.74
50%    46.93   48.06   57.38
75%    58.72   59.81   65.98
100%  100.00  100.00  100.00
```

```
mean    46.99    47.73    56.97
sd      18.78    16.73    13.29
var    352.50   280.00   176.70
NA's     0.00     0.00     0.00
N      517.00   517.00   517.00

$factors
NULL
```

```
m1poms <- lm(sal1 ~ satpoms + actpoms + ibspoms, data = dat2)
summary(m1poms)
```

```
Call:
lm(formula = sal1 ~ satpoms + actpoms + ibspoms, data = dat2)

Residuals:
   Min      1Q  Median      3Q     Max
-16069  -3697     184    2986   13705

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 11777.103    987.602  11.925  < 2e-16 ***
satpoms       102.879     18.737   5.491 6.31e-08 ***
actpoms        60.700     12.914   4.700 3.34e-06 ***
ibspoms        -6.723     14.500  -0.464    0.643
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4888 on 513 degrees of freedom
Multiple R^2: 0.1518,   Adjusted R^2: 0.1469
F-statistic: 30.61 on 3 and 513 DF,  p-value: < 2.2e-16
```

Oh, one more thing. Recall my point that partial and semi-partial correlations are completely worthless when 1) there is multicollinearity and 2) we are uncertain which variables should be in consideration. Notice how crazy your conclusions would be if you based them on the "full" model.

```
options(scipen = 10)
getPartialCor(m1all)
```

```
            sal1
sal1 -1.00000000
sat   0.08251460
act   0.14750834
ibs  -0.02019724
harv -0.07881873
```

```
getDeltaRsquare(m1all)
```

```
The deltaR-square values: the change in the R-square
      observed when a single term is removed.
Same as the square of the 'semi-partial correlation coefficient'
     deltaRsquare
sat  0.0057910045
act  0.0187893754
ibs  0.0003447358
harv 0.0052806872
```

```
options(scipen = 5)
```

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-9

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	5333.287	2887.95
	(2746.935)	(2604.482)
SAT	10.293*	8.742*
	(1.612)	(1.526)
ACT	189.013*	216.104*
	(51.491)	(48.488)
Iowa BS	-28.376	-9.273
	(26.647)	(25.199)
Major: Soc.	.	1756.4*
		(522.506)
Major: Nat.	.	4269.019*
		(513.778)
Prof. Parents: Yes	.	982.433*
		(460.044)
Parent Network: Yes	.	-297.101
		(455.358)
Gender: Male	.	481.538
		(425.234)
N	526	526
RMSE	5152.895	4829.95
R^2	0.144	0.255
adj R^2	0.139	0.244

$*p \leq 0.05$

```
m2small <- lm(sal2 ~ sat + act + ibs, data = dat)
m2all <- lm(sal2 ~ sat + act + ibs + major + pprof + pnet + gender, data = dat)
outreg(list(m2small, m2all), tight = TRUE, title = paste("Regression with sal2: Student-",i
    , sep=""),modelLabels = c("Test Scores Only","All Predictors"), varLabels = niceLabels,
     label = "table3")
```

Fancy T test. Lets use the big model to find out if $b_{pnetYES} = b_{pprofYES}$.

```
m2allc <- coef(m2all)
m2allv <- vcov(m2all)
numer <- m2allc["pprofYES"] - m2allc["pnetYES"]
names(numer) <- "difference"
denom <- sqrt(m2allv["pprofYES","pprofYES"] + m2allv["pnetYES","pnetYES"] - 2 * m2allv["
    pprofYES","pnetYES"])
print(paste("Fancy T: ", "Numerator = ", numer, "Denominator = ", denom))
```

```
[1] "Fancy T:  Numerator =  1279.53381668868 Denominator =  645.468929070692"
```

```
tval <- numer/denom
print("T ratio is")
```

```
[1] "T ratio is"
```

```
tval
```

```
difference
  1.982332
```

```
print("The two-tailed test would have p value")
```

```
[1] "The two-tailed test would have p value"
```

```
2 * pt(abs(tval), df = m2all$df, lower.tail = FALSE)
```

```
difference
0.04797105
```

Could I make a function that "just" gets that right and would I be damaging students by ruining their educational experience? This would be very easy if the output had the variable names "pprof" and "pnet", but because I've made them factors, they are now pprofYES and pnetYES, and thus either my function has to be clever or the user's have to be clever in naming their request.

```
fancyT <- function(model, parm1, parm2){
     mc <- coef(model)
     mv <- vcov(model)
     numer <- mc[parm1] - mc[parm2]
     denom <- sqrt(mv[parm1, parm1]
         + mv[parm2, parm2] - 2 * mv[parm1, parm2])
     tval <- numer/denom
     tdf <- model$df
     tvalp <- 2 * pt(abs(tval), df = tdf, lower.tail = FALSE)
   res <- c(numer, denom, tval, tdf, tvalp)
   names(res) <- c("parm1 - parm2", "SE(parm1 - parm2)", "T", "df", "p-value")
   res
 }
fancyT(m2all, parm1 = "pprofYES", parm2 = "pnetYES")
```

parm1 - parm2	SE(parm1 - parm2)	T	df	p-value
1279.53381669	645.46892907	1.98233216	517.00000000	0.04797105

```
m2all <- lm(sal2 ~ sat + act + ibs + major + pprof + pnet + gender, data = dat)
m2alldf <- model.frame(m2all)
m2small <- lm(sal2 ~ sat + act + ibs, data = m2alldf)
anova(m2small, m2all)
```

```
Analysis of Variance Table

Model 1: sal2 ~ sat + act + ibs
Model 2: sal2 ~ sat + act + ibs + major + pprof + pnet + gender
  Res.Df         RSS Df  Sum of Sq      F    Pr(>F)
1    522 13860315620
2    517 12060790300  5 1799525319 15.428 3.735e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nonlinear

```
nm1 <- lm(sal3 ~ harv + gender + major + pprof + pnet, data = dat)
nm2 <- lm(sal3 ~ log(harv) + gender + major + pprof + pnet, data = dat)
nm3 <- lm(sal3 ~ harv + I(harv*harv) + gender + major + pprof + pnet, data = dat)
library(rockchalk)
nd <- rockchalk::newdata(nm1, predVals = list(harv = plotSeq(dat$harv, 20)))
nd$m1fit <- predict(nm1, newdata = nd)
nd$m2fit <- predict(nm2, newdata = nd)
nd$m3fit <- predict(nm3, newdata = nd)
```

For the regression table, please see Table 4

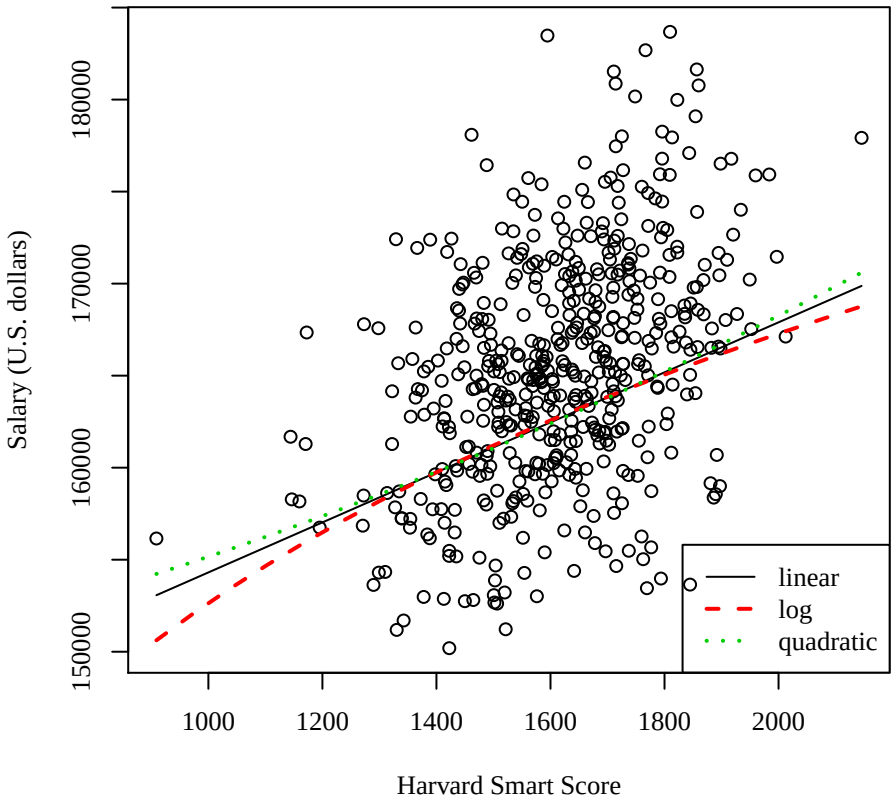
Table 4: Regression with sal3: Student-9

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	140720.296*	6731.796	147278.111*
	(2218.96)	(15781.463)	(13788.362)
Harvard SS	13.588*	.	5.279
	(1.363)		(17.296)
Gender: Male	-331.961	-334.542	-329.778
	(439.825)	(440.563)	(440.177)
Major: Soc.	1849.962*	1874.244*	1840.906*
	(538.819)	(539.612)	(539.549)
Major: Nat.	6270.16*	6286.285*	6265.898*
	(530.354)	(531.123)	(530.824)
Prof. Parents: Yes	939.144*	932.491	942.195*
	(474.429)	(475.237)	(474.826)
Parent Network: Yes	27.686	25.299	29.308
	(471.752)	(472.551)	(472.117)
ln(Harvard SS)	.	21122.461*	.
		(2140.716)	
Harvard SS^2	.	.	0.003
			(0.005)
N	521	521	521
RMSE	4977.24	4985.635	4980.961
R^2	0.358	0.356	0.358
adj R^2	0.35	0.348	0.349

$*p \leq 0.05$

```
outreg(list(nm1, nm2, nm3), tight = TRUE, title = paste("Regression with sal3: Student-",i,
    sep=""), modelLabels = c("Linear", "Log", "Quadratic"), varLabels = niceLabels, label
   = "table4")
```

```
plot(sal3 ~ harv, data = dat, xlab = "Harvard Smart Score", ylab = "Salary (U.S. dollars)")
lines(m1fit ~ harv, data = nd, lty = 1, col = 1)
lines(m2fit ~ harv, data = nd, lty = 2, col = 2, lwd = 2)
lines(m3fit ~ harv, data = nd, lty = 3, col = 3, lwd = 2)
legend("bottomright", legend = c("linear", "log", "quadratic"), lty = c(1,2,3), col = c
    (1,2,3), lwd = c(1,2,2))
```




```
cm1 <- lm(sal2 ~ major, data = dat)
dat$major2  <- relevel(dat$major, ref = "S")
cm2 <- lm(sal2 ~ major2, data = dat)
cm3 <- lm(sal2 ~ sat + act + ibs + major + pprof + pnet + gender, data = dat)
cm4 <- lm(sal2 ~ sat + act + ibs + major2 + pprof + pnet + gender, data = dat)
```

```
outreg(list(cm1, cm2, cm3, cm4), tight = TRUE, title = paste("Categorical Regressions:
    Student-", i, sep=""), modelLabels = c("major", "major2", "major full", "major2 full"),
     varLabels = niceLabels)
```

```
predictOMatic(cm1)
```

```
$major
            fit major
H (40%) 20868.51     H
N (30%) 25396.33     N
S (30%) 23023.18     S

attr(,"flnames")
[1] "major"
```

```
predictOMatic(cm2)
```

```
$major2
            fit major2
H (40%) 20868.51      H
N (30%) 25396.33      N
S (30%) 23023.18      S

attr(,"flnames")
[1] "major2"
```

Table 5: Categorical Regressions: Student-9

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	20868.511* (369.537)	23023.178* (396.722)	2887.95 (2604.482)	4644.35 (2623.229)
Major: Soc.	2154.667* (542.168)	.	1756.4* (522.506)	.
Major: Nat.	4527.82* (530.707)	.	4269.019* (513.778)	.
Major 2: Hum.	.	-2154.667* (542.168)	.	-1756.4* (522.506)
Major 2: Nat.	.	2373.153* (549.982)	.	2512.619* (528.587)
SAT	.	.	8.742* (1.526)	8.742* (1.526)
ACT	.	.	216.104* (48.488)	216.104* (48.488)
Iowa BS	.	.	-9.273 (25.199)	-9.273 (25.199)
Prof. Parents: Yes	.	.	982.433* (460.044)	982.433* (460.044)
Parent Network: Yes	.	.	-297.101 (455.358)	-297.101 (455.358)
Gender: Male	.	.	481.538 (425.234)	481.538 (425.234)
N	573	573	526	526
RMSE	5278.043	5278.043	4829.95	4829.95
R^2	0.113	0.113	0.255	0.255
adj R^2	0.11	0.11	0.244	0.244

$*p \leq 0.05$