Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 19
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%	5.56	1140.00	68.60	5813.00	8880.00	147200.00	1122.00
25%	18.25	1510.00	93.19	16920.00	19660.00	161700.00	1489.00
50%	21.80	1616.00	99.73	20590.00	23280.00	165200.00	1594.00
75%	25.20	1736.00	105.90	24030.00	27450.00	169100.00	1714.00
100%	36.51	2153.00	140.20	39740.00	44580.00	180700.00	2118.00
mean	21.84	1621.00	99.41	20520.00	23500.00	165300.00	1600.00
sd	5.00	159.50	9.80	5318.00	5909.00	5757.00	157.20
var	24.96	25430.00	96.05	28280000.00	34920000.00	33140000.00	24710.00
NA's	17.00	54.00	0.00	7.00	0.00	0.00	38.00
N	544.00	544.00	544.00	544.00	544.00	544.00	544.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
           gender                        major                        pnet
M             :290.0000   S             :189.0000   NO            :383.0000
F             :254.0000   N             :186.0000   YES           :161.0000
NA's          :  0.0000   H             :169.0000   NA's          :  0.0000
entropy       :  0.9968   NA's          :  0.0000   entropy       :  0.8763
normedEntropy:  0.9968    entropy       :  1.5832   normedEntropy:  0.8763
N             :544.0000   normedEntropy:  0.9989    N             :544.0000
                          N             :544.0000
           pprof
NO            :394.0000
YES           :150.0000
NA's          :  0.0000
entropy       :  0.8496
normedEntropy:  0.8496
N             :544.0000
```

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: 0x2819738>
act ~ sat + ibs + harv
<environment: 0x2819738>
ibs ~ sat + act + harv
<environment: 0x2819738>
harv ~ sat + act + ibs
<environment: 0x2819738>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998416 0.8711838 0.2607309 0.9998458
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
6312.756549    7.763000    1.352687 6485.084552
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.43 0.43 1.00
act  0.43 1.00 0.42 0.45
ibs  0.43 0.42 1.00 0.44
harv 1.00 0.45 0.44 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-19

	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)	2165.779	12915.048*	9047.347*	1817.868	1192.666	1751.901
	(2276.493)	(983.434)	(2287.848)	(2298.557)	(2836.963)	(2693.797)
SAT	11.456*	.	.	.	99.104	8.38*
	(1.415)				(118.103)	(1.631)
ACT	.	349.894*	.	.	302.297*	213.92*
		(43.858)			(135.297)	(53.27)
Iowa BS	.	.	115.454*	.	14.039	7.163
			(22.921)		(27.592)	(26.212)
Harvard SS	.	.	.	11.488*	-90.809	.
				(1.411)	(118.093)	
N	499	521	537	485	439	486
RMSE	4984.629	5010.328	5200.659	4962.039	4925.354	4887.097
R^2	0.116	0.109	0.045	0.121	0.15	0.152
adj R^2	0.115	0.108	0.043	0.119	0.142	0.147

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    436 1.0551e+10
2    434 1.0528e+10  2  22345964 0.4606 0.6312
```

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)	2183.298	13066.129*	9497.326*	1918.477	1192.666	1751.901
	(2282.932)	(1053.211)	(2392.428)	(2425.832)	(2836.963)	(2693.797)
SAT	11.491*	.	.	.	99.104	8.38*
	(1.419)				(118.103)	(1.631)
ACT	.	341.878*	.	.	302.297*	213.92*
		(46.828)			(135.297)	(53.27)
Iowa BS	.	.	111.394*	.	14.039	7.163
			(23.945)		(27.592)	(26.212)
Harvard SS	.	.	.	11.45*	-90.809	.
				(1.488)	(118.093)	
N	486	486	486	439	439	486
RMSE	4969.674	5025.898	5180.855	4996.097	4925.354	4887.097
R^2	0.119	0.099	0.043	0.119	0.15	0.152
adj R^2	0.117	0.097	0.041	0.117	0.142	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   0.22782609
act   0.17992730
ibs   0.01244529
```

```
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 0.0464364087
act 0.0283785512
ibs 0.0001313957
```

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%     41.27   34.19   36.80
50%     52.79   43.28   47.33
75%     63.65   52.30   59.71
100%   100.00  100.00  100.00
```

```
mean    52.98    43.04    48.01
sd      15.75    13.72    15.96
var    247.90   188.10   254.90
NA's     0.00     0.00     0.00
N      486.00   486.00   486.00

$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
-12933   -3151     109    3397   16311

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 12837.714    920.783  13.942  < 2e-16 ***
satpoms        83.447     16.245   5.137 4.06e-07 ***
actpoms        66.208     16.487   4.016 6.87e-05 ***
ibspoms         5.131     18.776   0.273    0.785
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4887 on 482 degrees of freedom
Multiple R^2: 0.1518,  Adjusted R^2: 0.1465
F-statistic: 28.75 on 3 and 482 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   0.04024687
act   0.10663907
ibs   0.02441659
harv -0.03688622
```

```
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  0.0013790247
act  0.0097769522
ibs  0.0005070291
harv 0.0011580395
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-19

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	2773.472 (3038.497)	1245.183 (2695.331)
SAT	8.619* (1.839)	8.702* (1.624)
ACT	197.272* (59.837)	205.634* (52.734)
Iowa BS	26.445 (29.367)	6.156 (25.898)
Major: Soc.	.	2076.183* (539.997)
Major: Nat.	.	6339.727* (546.55)
Prof. Parents: Yes	.	648.859 (493.183)
Parent Network: Yes	.	1102.523* (482.728)
Gender: Male	.	-246.391 (440.101)
N	492	492
RMSE	5532.033	4866.62
R^2	0.13	0.333
adj R^2	0.124	0.322

$*p \leq 0.05$

```
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 =  -453.66432519283 Denominator =  687.628665646606"
```

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

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

```
tval
```

```
difference
-0.6597519
```

```
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
 0.5097275
```

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
     -453.6643252       687.6286656        -0.6597519       483.0000000         0.5097275
```

```
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    488 14934455538
2    483 11439367954  5 3495087584 29.514 < 2.2e-16 ***
---
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

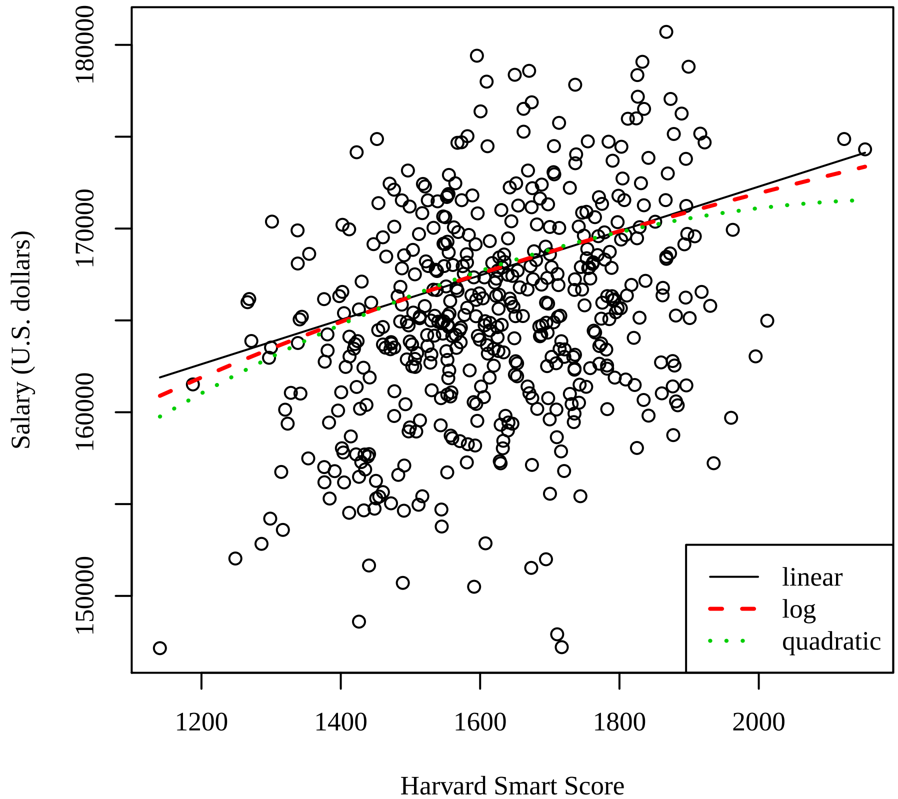
Table 4: Regression with sal3: Student-19

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	142141.104*	16683.285	115597.043*
	(2363.392)	(16923.458)	(17565.724)
Harvard SS	12.085*	.	45.062*
	(1.428)		(21.672)
Gender: Male	532.348	532.547	528.211
	(454.571)	(453.811)	(453.957)
Major: Soc.	2686.307*	2692.452*	2700.376*
	(564.329)	(563.367)	(563.632)
Major: Nat.	5442.101*	5442.594*	5442.493*
	(559.489)	(558.552)	(558.723)
Prof. Parents: Yes	1066.617*	1055.675*	1050.957*
	(509.07)	(508.281)	(508.476)
Parent Network: Yes	368.316	376.59	383.124
	(496.64)	(495.846)	(496.055)
ln(Harvard SS)	.	19638.06*	.
		(2291.366)	
Harvard SS^2	.	.	-0.01
			(0.007)
N	490	490	490
RMSE	5015.893	5007.509	5009.025
R^2	0.268	0.27	0.272
adj R^2	0.259	0.261	0.261

$*p \leq 0.05$

```
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
S (30%) 22882.49     S
N (30%) 26821.07     N
H (30%) 20530.54     H

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

```
predictOMatic(cm2)
```

```
$major2
             fit major2
S (30%) 22882.49      S
N (30%) 26821.07      N
H (30%) 20530.54      H

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

Table 5: Categorical Regressions: Student-19

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	20530.544*	22882.491*	1245.183	3321.366
	(409.694)	(387.411)	(2695.331)	(2709.868)
Major: Soc.	2351.946*	.	2076.183*	.
	(563.858)		(539.997)	
Major: Nat.	6290.528*	.	6339.727*	.
	(566.001)		(546.55)	
Major 2: Hum.	.	-2351.946*	.	-2076.183*
		(563.858)		(539.997)
Major 2: Nat.	.	3938.581*	.	4263.544*
		(550.086)		(531.055)
SAT	.	.	8.702*	8.702*
			(1.624)	(1.624)
ACT	.	.	205.634*	205.634*
			(52.734)	(52.734)
Iowa BS	.	.	6.156	6.156
			(25.898)	(25.898)
Prof. Parents: Yes	.	.	648.859	648.859
			(493.183)	(493.183)
Parent Network: Yes	.	.	1102.523*	1102.523*
			(482.728)	(482.728)
Gender: Male	.	.	-246.391	-246.391
			(440.101)	(440.101)
N	544	544	492	492
RMSE	5326.018	5326.018	4866.62	4866.62
R^2	0.191	0.191	0.333	0.333
adj R^2	0.188	0.188	0.322	0.322

$*p \leq 0.05$