Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 1
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%	0.97	1076.00	61.62	2031.00	6226.00	147400.00	1066.00
25%	18.84	1518.00	93.31	16790.00	19480.00	162000.00	1502.00
50%	22.01	1625.00	99.81	20280.00	23340.00	165800.00	1607.00
75%	25.64	1739.00	106.40	23940.00	27060.00	169500.00	1725.00
100%	39.62	2093.00	128.20	36830.00	40620.00	183900.00	2069.00
mean	21.93	1624.00	99.78	20280.00	23230.00	165700.00	1607.00
sd	5.17	168.70	9.78	5539.00	5809.00	5687.00	167.30
var	26.69	28450.00	95.55	30680000.00	33740000.00	32340000.00	28000.00
NA's	14.00	51.00	0.00	12.00	0.00	0.00	27.00
N	581.00	581.00	581.00	581.00	581.00	581.00	581.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                    major                     pnet
M             :301.0000   H             :201.0000   NO            :412.0000
F             :280.0000   N             :196.0000   YES           :169.0000
NA's          :  0.0000   S             :184.0000   NA's          :  0.0000
entropy       :  0.9991   NA's          :  0.0000   entropy       :  0.8699
normedEntropy:  0.9991   entropy       :  1.5840   normedEntropy:  0.8699
N             :581.0000   normedEntropy:  0.9994   N             :581.0000
                          N             :581.0000
          pprof
NO            :420.0000
YES           :161.0000
NA's          :  0.0000
entropy       :  0.8515
normedEntropy:  0.8515
N             :581.0000
```

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: 0x1f89840>
act ~ sat + ibs + harv
<environment: 0x1f89840>
ibs ~ sat + act + harv
<environment: 0x1f89840>
harv ~ sat + act + ibs
<environment: 0x1f89840>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998652 0.8819047 0.3325010 0.9998687
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
7415.907036    8.467738    1.498130 7614.177626
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.43 0.49 1.00
act  0.43 1.00 0.48 0.46
ibs  0.49 0.48 1.00 0.50
harv 1.00 0.46 0.50 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-1

	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)	-366.328	12556.995*	7000.211*	-205.302	-1.541	-346.937
	(2127.037)	(969.461)	(2313.103)	(2163.761)	(2690.013)	(2575.122)
SAT	12.846*	.	.	.	121.222	10.133*
	(1.316)				(118.58)	(1.557)
ACT	.	352.909*	.	.	327.524*	214.06*
		(43.061)			(128.113)	(49.841)
Iowa BS	.	.	133.11*	.	-3.602	-3.175
			(23.082)		(28.683)	(27.247)
Harvard SS	.	.	.	12.587*	-111.333	.
				(1.325)	(118.502)	
N	542	555	569	519	481	528
RMSE	5140.508	5268.725	5388.089	5102.134	5090.153	5089.195
R^2	0.15	0.108	0.055	0.149	0.182	0.181
adj R^2	0.148	0.107	0.054	0.147	0.175	0.177

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    478 1.2356e+10
2    476 1.2333e+10  2   2.3e+07 0.4439 0.6418
```

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)	-362.592	12628.722*	6912.591*	-350.135	-1.541	-346.937
	(2161.137)	(996.495)	(2409.355)	(2252.864)	(2690.013)	(2575.122)
SAT	12.859*	.	.	.	121.222	10.133*
	(1.338)				(118.58)	(1.557)
ACT	.	350.586*	.	.	327.524*	214.06*
		(44.345)			(128.113)	(49.841)
Iowa BS	.	.	134.176*	.	-3.602	-3.175
			(24.044)		(28.683)	(27.247)
Harvard SS	.	.	.	12.675*	-111.333	.
				(1.38)	(118.502)	
N	528	528	528	481	481	528
RMSE	5177.412	5307.076	5454.397	5172.095	5090.153	5089.195
R^2	0.149	0.106	0.056	0.15	0.182	0.181
adj R^2	0.148	0.105	0.054	0.148	0.175	0.177

$*p \leq 0.05$

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   0.27342264
act   0.18440391
ibs  -0.00508998
```

```
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 0.0661584731
act 0.0288228043
ibs 0.0000212136
```

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%    45.98   47.34   43.14
50%    54.36   57.31   54.06
75%    63.68   67.27   65.97
100%  100.00  100.00  100.00
```

```
mean    54.05    57.18    53.84
sd      13.49    14.83    16.80
var    181.90   220.00   282.40
NA's     0.00     0.00     0.00
N      528.00   528.00   528.00

$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
-14800.9  -3637.5    234.5   3230.7  16023.6

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 10469.689   1060.280   9.874  < 2e-16 ***
satpoms       101.638     15.620   6.507 1.80e-10 ***
actpoms        82.734     19.264   4.295 2.08e-05 ***
ibspoms        -2.115     18.155  -0.117    0.907
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5089 on 524 degrees of freedom
Multiple R^2: 0.1812,   Adjusted R^2: 0.1765
F-statistic: 38.66 on 3 and 524 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 -1.000000000
sat   0.046804992
act   0.116381837
ibs  -0.005755627
harv -0.043022031
```

```
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  0.00179670006
act  0.01123651915
ibs  0.00002711056
harv 0.00151748769
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-1

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	5333.186	-404.7
	(2751.876)	(2645.846)
SAT	8.094*	9.53*
	(1.646)	(1.545)
ACT	211.678*	207.874*
	(53.137)	(49.52)
Iowa BS	2.693	6.153
	(28.808)	(26.894)
Major: Soc.	.	2491.415*
		(539.325)
Major: Nat.	.	4467.718*
		(529.824)
Prof. Parents: Yes	.	1532.37*
		(488.068)
Parent Network: Yes	.	1407.441*
		(489.095)
Gender: Male	.	129.227
		(438.49)
N	540	540
RMSE	5466.118	5077.386
R^2	0.128	0.255
adj R^2	0.123	0.243

$*p \leq 0.05$

```
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 =  124.929756329568 Denominator =  694.393678577456"
```

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

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

```
tval
```

```
difference
  0.179912
```

```
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
 0.8572903
```

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
     124.9297563       694.3936786         0.1799120       531.0000000         0.8572903
```

```
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    536 16014845864
2    531 13689099966  5 2325745899 18.043 < 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

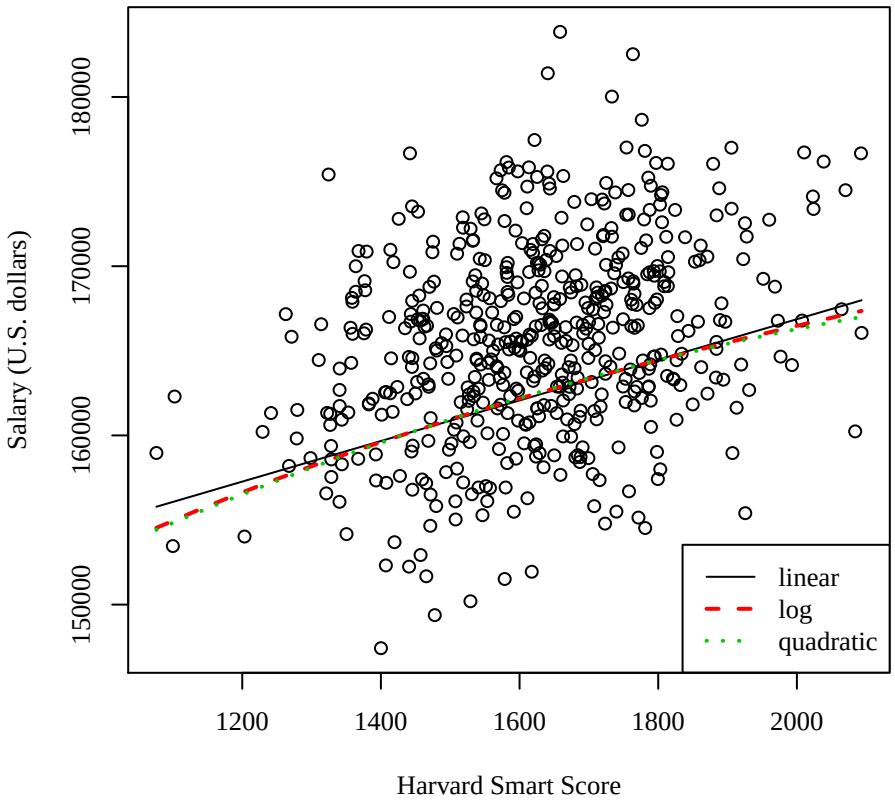
Table 4: Regression with sal3: Student-1

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	143776.252*	20903.382	130508.105*
	(2168.538)	(15177.682)	(13737.746)
Harvard SS	11.999*	.	28.509
	(1.284)		(16.929)
Gender: Male	-905.461*	-880.224*	-873.049*
	(429.472)	(429.197)	(430.766)
Major: Soc.	2219.431*	2258.695*	2272.806*
	(527.234)	(526.67)	(530.072)
Major: Nat.	5513.809*	5521.088*	5522.104*
	(520.266)	(519.847)	(520.357)
Prof. Parents: Yes	795.005	777.926	774.061
	(482.365)	(481.836)	(482.86)
Parent Network: Yes	153.919	153.487	150.597
	(469.653)	(469.223)	(469.685)
ln(Harvard SS)	.	19267.994*	.
		(2048.82)	
Harvard SS^2	.	.	-0.005
			(0.005)
N	530	530	530
RMSE	4930.654	4926.269	4930.858
R^2	0.268	0.269	0.269
adj R^2	0.26	0.261	0.26

$*p \leq 0.05$

```
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 (30%) 20976.51     H
N (30%) 25152.69     N
S (30%) 23647.35     S

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
H (30%) 20976.51      H
N (30%) 25152.69      N
S (30%) 23647.35      S

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

Table 5: Categorical Regressions: Student-1

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	20976.505* (391.346)	23647.35* (409.025)	-404.7 (2645.846)	2086.714 (2659.752)
Major: Soc.	2670.845* (566.086)	.	2491.415* (539.325)	.
Major: Nat.	4176.184* (556.965)	.	4467.718* (529.824)	.
Major 2: Hum.	.	-2670.845* (566.086)	.	-2491.415* (539.325)
Major 2: Nat.	.	1505.339* (569.526)	.	1976.304* (551.936)
SAT	.	.	9.53* (1.545)	9.53* (1.545)
ACT	.	.	207.874* (49.52)	207.874* (49.52)
Iowa BS	.	.	6.153 (26.894)	6.153 (26.894)
Prof. Parents: Yes	.	.	1532.37* (488.068)	1532.37* (488.068)
Parent Network: Yes	.	.	1407.441* (489.095)	1407.441* (489.095)
Gender: Male	.	.	129.227 (438.49)	129.227 (438.49)
N	581	581	540	540
RMSE	5548.284	5548.284	5077.386	5077.386
R^2	0.091	0.091	0.255	0.255
adj R^2	0.088	0.088	0.243	0.243

$*p \leq 0.05$