Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 12
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%	4.89	1075.00	56.49	-3.73	1112.00	145000.00	1068.00
25%	18.38	1504.00	92.92	16390.00	18620.00	161400.00	1485.00
50%	21.98	1610.00	100.10	19950.00	22750.00	165200.00	1593.00
75%	25.49	1713.00	106.70	23180.00	26900.00	169800.00	1699.00
100%	36.45	2194.00	131.90	39190.00	42340.00	181800.00	2166.00
mean	21.92	1607.00	99.77	19840.00	22800.00	165300.00	1591.00
sd	4.97	163.70	10.52	5313.00	5946.00	6166.00	162.80
var	24.71	26810.00	110.70	28230000.00	35360000.00	38020000.00	26500.00
NA's	16.00	61.00	0.00	15.00	0.00	0.00	29.00
N	562.00	562.00	562.00	562.00	562.00	562.00	562.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                  major                   pnet                   pprof
M             :291.0000   S             :192.0000   NO            :400.0000   NO
    :394.00
F             :271.0000   N             :192.0000   YES           :162.0000   YES
    :168.00
NA's          :  0.0000   H             :178.0000   NA's          :  0.0000   NA's          :
    0.00
entropy       :  0.9991   NA's          :  0.0000   entropy       :  0.8665   entropy       :
    0.88
normedEntropy:  0.9991   entropy       :  1.5841   normedEntropy:  0.8665   normedEntropy:
    0.88
N             :562.0000   normedEntropy:  0.9994   N             :562.0000   N
    :562.00
                          N             :562.0000
```

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: 0x27ed030>
act ~ sat + ibs + harv
<environment: 0x27ed030>
ibs ~ sat + act + harv
<environment: 0x27ed030>
harv ~ sat + act + ibs
<environment: 0x27ed030>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998581 0.8703402 0.2510563 0.9998619
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
7045.833487    7.712493    1.335214 7239.128432
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.46 0.42 1.00
act  0.46 1.00 0.43 0.48
ibs  0.42 0.43 1.00 0.43
harv 1.00 0.48 0.43 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-12

	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)	-1075.752	11825.684*	8566.587*	-1832.47	-2923.432	-2237.105
	(2140.697)	(986.79)	(2134.486)	(2184.953)	(2679.663)	(2520.763)
SAT	13.075*	.	.	.	91.373	9.815*
	(1.336)				(117.423)	(1.563)
ACT	.	364.174*	.	.	253.711*	190.844*
		(43.959)			(125.858)	(51.553)
Iowa BS	.	.	112.962*	.	15.217	21.614
			(21.279)		(25.579)	(23.828)
Harvard SS	.	.	.	13.406*	-80.496	.
				(1.35)	(117.323)	
N	519	532	547	489	448	504
RMSE	4893.553	5009.826	5185.55	4838.1	4775.389	4823.337
R^2	0.156	0.115	0.049	0.168	0.2	0.186
adj R^2	0.155	0.113	0.047	0.167	0.193	0.181

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    445 1.0123e+10
2    443 1.0102e+10  2  20312620 0.4454 0.6409
```

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)	-1094.275	11938.756*	7209.222*	-2348.058	-2923.432	-2237.105
	(2175.237)	(1020.192)	(2206.491)	(2254.844)	(2679.663)	(2520.763)
SAT	13.066*	.	.	.	91.373	9.815*
	(1.358)				(117.423)	(1.563)
ACT	.	356.861*	.	.	253.711*	190.844*
		(45.606)			(125.858)	(51.553)
Iowa BS	.	.	125.346*	.	15.217	21.614
			(21.975)		(25.579)	(23.828)
Harvard SS	.	.	.	13.658*	-80.496	.
				(1.394)	(117.323)	
N	504	504	504	448	448	504
RMSE	4901.293	5035.726	5169.112	4825.927	4775.389	4823.337
R^2	0.156	0.109	0.061	0.177	0.2	0.186
adj R^2	0.154	0.107	0.059	0.175	0.193	0.181

$*p \leq 0.05$

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   0.27034380
act   0.16333160
ibs   0.04053272
```

```
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  0.064217086
act  0.022322400
ibs  0.001340241
```

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%    42.73   48.78   38.11
50%    53.98   57.81   47.82
75%    64.77   66.56   57.31
100%  100.00  100.00  100.00
```

```
mean    53.65    57.49    47.85
sd      15.60    13.90    14.66
var    243.30   193.30   214.90
NA's     0.00     0.00     0.00
N      504.00   504.00   504.00

$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
-13693.0  -3253.7    -80.4   3290.5  16622.9

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 10402.13    1022.66  10.172  < 2e-16 ***
satpoms       107.73      17.16   6.279 7.41e-10 ***
actpoms        60.23      16.27   3.702 0.000238 ***
ibspoms        16.31      17.98   0.907 0.364799
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4823 on 500 degrees of freedom
Multiple R^2: 0.1856,   Adjusted R^2: 0.1807
F-statistic: 37.97 on 3 and 500 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   0.03694582
act   0.09533971
ibs   0.02825309
harv -0.03258054
```

```
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  0.0010937917
act  0.0073404725
ibs  0.0006392776
harv 0.0008503318
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-12

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	-952.172	-3645.332
	(2759.355)	(2509.568)
SAT	10.849*	10.023*
	(1.724)	(1.556)
ACT	235.283*	195.481*
	(56.73)	(51.329)
Iowa BS	13.041	27.862
	(26.299)	(23.83)
Major: Soc.	.	2732.315*
		(536.856)
Major: Nat.	.	5636.531*
		(534.784)
Prof. Parents: Yes	.	915.419
		(468.282)
Parent Network: Yes	.	1198.107*
		(475.563)
Gender: Male	.	-305.987
		(432.019)
N	517	517
RMSE	5392.638	4860.884
R^2	0.187	0.346
adj R^2	0.182	0.335

$*p \leq 0.05$

```
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 =  -282.687149973692 Denominator =  656.182884677066"
```

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

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

```
tval
```

```
difference
-0.4308054
```

```
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
 0.6667926
```

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
-282.6871500	656.1828847	-0.4308054	508.0000000	0.6667926

```
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    513 14918318056
2    508 12003120764  5 2915197292 24.676 < 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

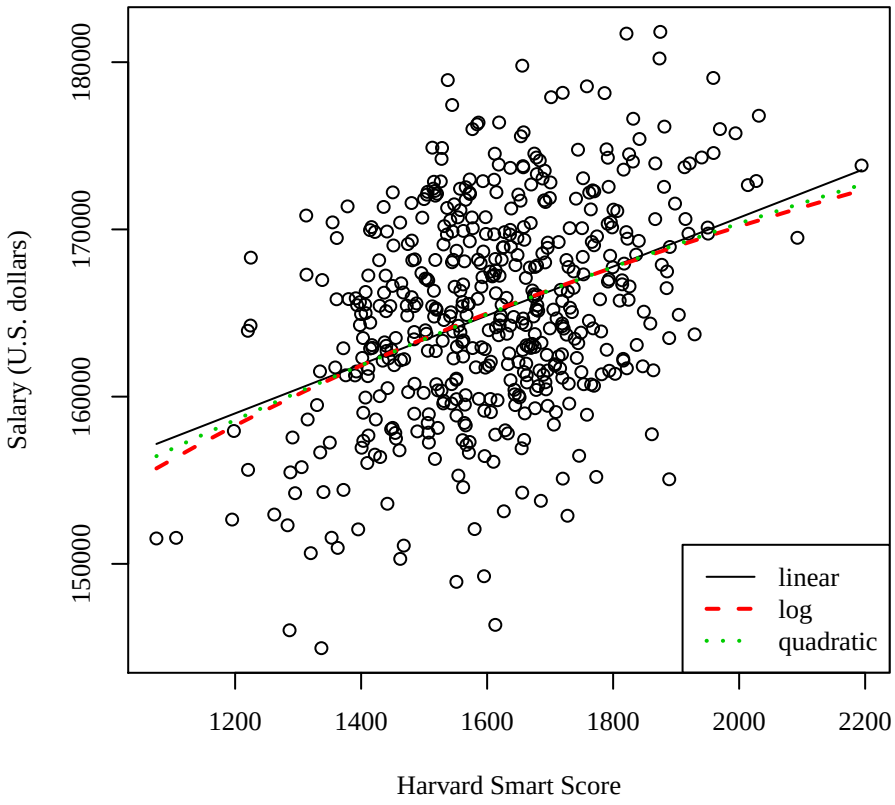
Table 4: Regression with sal3: Student-12

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	139442.689*	-9006.741	132266.269*
	(2387.734)	(17154.295)	(15686.823)
Harvard SS	14.631*	.	23.627
	(1.466)		(19.492)
Gender: Male	-159.257	-169.453	-166.412
	(481.463)	(481.207)	(482.094)
Major: Soc.	2159.45*	2173.428*	2167.018*
	(588.416)	(587.978)	(589.111)
Major: Nat.	5234.097*	5208.189*	5219.98*
	(595.08)	(594.872)	(596.334)
Prof. Parents: Yes	308.407	320.053	316.291
	(531.332)	(531.028)	(532.028)
Parent Network: Yes	-821.76	-796.258	-805.13
	(533.351)	(533.006)	(534.984)
ln(Harvard SS)	.	23310.626*	.
		(2328.41)	
Harvard SS^2	.	.	-0.003
			(0.006)
N	501	501	501
RMSE	5354.461	5351.349	5358.724
R^2	0.277	0.278	0.278
adj R^2	0.268	0.269	0.267

$*p \leq 0.05$

```
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%) 22735.33     S
N (30%) 25667.47     N
H (30%) 19765.80     H

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
S (30%) 22735.33      S
N (30%) 25667.47      N
H (30%) 19765.80      H

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

Table 5: Categorical Regressions: Student-12

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	19765.797* (408.67)	22735.327* (393.488)	-3645.332 (2509.568)	-913.017 (2528.473)
Major: Soc.	2969.53* (567.313)	.	2732.315* (536.856)	.
Major: Nat.	5901.669* (567.313)	.	5636.531* (534.784)	.
Major 2: Hum.	.	-2969.53* (567.313)	.	-2732.315* (536.856)
Major 2: Nat.	.	2932.139* (556.476)	.	2904.216* (513.301)
SAT	.	.	10.023* (1.556)	10.023* (1.556)
ACT	.	.	195.481* (51.329)	195.481* (51.329)
Iowa BS	.	.	27.862 (23.83)	27.862 (23.83)
Prof. Parents: Yes	.	.	915.419 (468.282)	915.419 (468.282)
Parent Network: Yes	.	.	1198.107* (475.563)	1198.107* (475.563)
Gender: Male	.	.	-305.987 (432.019)	-305.987 (432.019)
N	562	562	517	517
RMSE	5452.333	5452.333	4860.884	4860.884
R^2	0.162	0.162	0.346	0.346
adj R^2	0.159	0.159	0.335	0.335

$*p \leq 0.05$