Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 13
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%	6.18	1091.00	72.02	3306.00	5317.00	146900.00	1075.00
25%	18.60	1518.00	93.45	16640.00	19510.00	161000.00	1495.00
50%	21.84	1630.00	100.10	20250.00	23050.00	165400.00	1606.00
75%	25.26	1723.00	106.30	23830.00	26960.00	169400.00	1699.00
100%	37.88	2102.00	122.80	40980.00	42920.00	182100.00	2072.00
mean	21.91	1625.00	100.20	20210.00	23260.00	165300.00	1600.00
sd	5.08	163.20	9.38	5429.00	5825.00	5947.00	159.00
var	25.82	26650.00	88.04	29480000.00	33930000.00	35370000.00	25300.00
NA's	10.00	46.00	0.00	8.00	0.00	0.00	19.00
N	524.00	524.00	524.00	524.00	524.00	524.00	524.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                   major                    pnet
M             :267.0000   S             :180.0000   NO            :351.0000
F             :257.0000   N             :180.0000   YES           :173.0000
NA's          :  0.0000   H             :164.0000   NA's          :  0.0000
entropy       :  0.9997   NA's          :  0.0000   entropy       :  0.9151
normedEntropy:  0.9997   entropy       :  1.5836   normedEntropy:  0.9151
N             :524.0000   normedEntropy:  0.9991   N             :524.0000
                          N             :524.0000
          pprof
NO            :377.0000
YES           :147.0000
NA's          :  0.0000
entropy       :  0.8562
normedEntropy:  0.8562
N             :524.0000
```

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: 0x1c8dd78>
act ~ sat + ibs + harv
<environment: 0x1c8dd78>
ibs ~ sat + act + harv
<environment: 0x1c8dd78>
harv ~ sat + act + ibs
<environment: 0x1c8dd78>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998396 0.8645667 0.1707067 0.9998437
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
6233.535254    7.383710    1.205846 6399.658291
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.42 0.37 1.00
act  0.42 1.00 0.32 0.45
ibs  0.37 0.32 1.00 0.37
harv 1.00 0.45 0.37 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-13

	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)	-851.365	12385.127*	7862.561*	-757.943	-1014.502	-2446.509
	(2269.573)	(1011.599)	(2515.193)	(2303.644)	(3049.667)	(2853.87)
SAT	13.102*	.	.	.	47.138	9.97*
	(1.411)				(117.41)	(1.598)
ACT	.	356.738*	.	.	248.136	207.006*
		(44.918)			(126.965)	(49.345)
Iowa BS	.	.	123.24*	.	13.841	20.739
			(24.999)		(28.119)	(26.127)
Harvard SS	.	.	.	12.96*	-37.625	.
				(1.41)	(117.482)	
N	497	506	516	472	444	487
RMSE	5027.481	5134.069	5310.432	5022.508	5009.134	4949.21
R^2	0.148	0.111	0.045	0.152	0.171	0.182
adj R^2	0.147	0.109	0.043	0.151	0.164	0.177

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    441 1.1024e+10
2    439 1.1015e+10  2   9217139 0.1837 0.8323
```

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)	-859.699	12501.946*	8552.518*	-361.182	-1014.502	-2446.509
	(2301.96)	(1036.675)	(2611.101)	(2430.966)	(3049.667)	(2853.87)
SAT	13.112*	.	.	.	47.138	9.97*
	(1.432)				(117.41)	(1.598)
ACT	.	347.657*	.	.	248.136	207.006*
		(46.093)			(126.965)	(49.345)
Iowa BS	.	.	115.451*	.	13.841	20.739
			(25.95)		(28.119)	(26.127)
Harvard SS	.	.	.	12.682*	-37.625	.
				(1.49)	(117.482)	
N	487	487	487	444	444	487
RMSE	5041.221	5165.25	5351.674	5082.908	5009.134	4949.21
R^2	0.147	0.105	0.039	0.141	0.171	0.182
adj R^2	0.146	0.103	0.037	0.139	0.164	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   0.27304820
act   0.18749814
ibs   0.03609434
```

```
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  0.065925536
act  0.029816874
ibs  0.001067505
```

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%    38.96   42.51   44.13
50%    49.31   55.53   55.84
75%    60.16   68.02   65.83
100%  100.00  100.00  100.00
```

```
mean    49.62    55.50    55.17
sd      16.04    18.43    16.78
var    257.10   339.80   281.50
NA's     0.00     0.00     0.00
N      487.00   487.00   487.00

$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
-17145.1  -3147.9    -33.4   3231.8  18075.1

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 11042.24     956.79  11.541  < 2e-16 ***
satpoms        94.92      15.22   6.238 9.70e-10 ***
actpoms        65.62      15.64   4.195 3.25e-05 ***
ibspoms        10.53      13.26   0.794    0.428
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4949 on 483 degrees of freedom
Multiple R^2: 0.1817,   Adjusted R^2: 0.1766
F-statistic: 35.74 on 3 and 483 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   0.01915838
act   0.09287380
ibs   0.02348534
harv -0.01528373
```

```
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  0.0003043097
act  0.0072108637
ibs  0.0004573744
harv 0.0001936416
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-13

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	1062.828	-1644.587
	(3098.671)	(2908.426)
SAT	10.969*	10.088*
	(1.738)	(1.597)
ACT	139.483*	191.452*
	(53.493)	(49.55)
Iowa BS	15.257	20.802
	(28.293)	(26.003)
Major: Soc.	.	1598.689*
		(556.034)
Major: Nat.	.	5154.568*
		(554.871)
Prof. Parents: Yes	.	802.741
		(494.745)
Parent Network: Yes	.	509.602
		(472.524)
Gender: Male	.	-586.754
		(448.043)
N	495	495
RMSE	5408.643	4959.472
R^2	0.142	0.286
adj R^2	0.136	0.274

$*p \leq 0.05$

```
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 =  293.139253079534 Denominator =  683.094762372726"
```

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

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

```
tval
```

```
difference
 0.4291341
```

```
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
 0.6680158
```

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
     293.1392531       683.0947624         0.4291341       486.0000000         0.6680158
```

```
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    491 14363426575
2    486 11953831023  5 2409595552 19.593 < 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

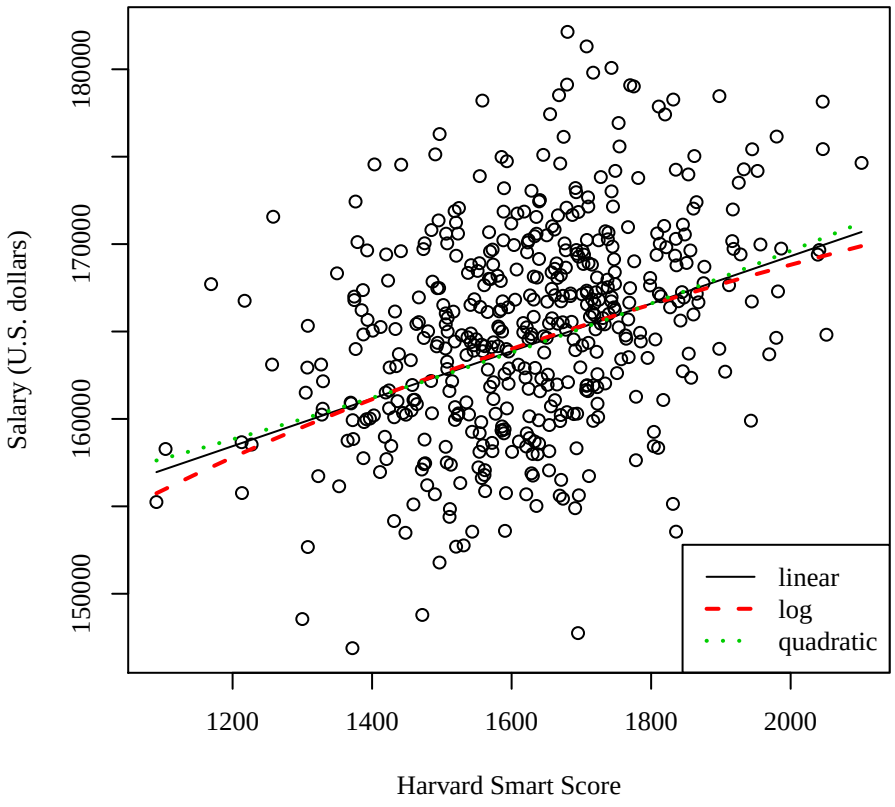
Table 4: Regression with sal3: Student-13

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	141107.047*	3853.509	148026.754*
	(2410.622)	(16984.744)	(15477.941)
Harvard SS	13.59*	.	4.987
	(1.434)		(19.062)
Gender: Male	-559.169	-549.126	-566.344
	(468.468)	(469.147)	(469.132)
Major: Soc.	1585.096*	1610.27*	1565.781*
	(570.257)	(571.084)	(572.332)
Major: Nat.	4836.145*	4861.404*	4820.294*
	(581.891)	(582.662)	(583.435)
Prof. Parents: Yes	1652.86*	1644.442*	1657.201*
	(525.737)	(526.416)	(526.269)
Parent Network: Yes	-804.913	-800.869	-807.484
	(497.162)	(497.837)	(497.615)
ln(Harvard SS)	.	21563.741*	.
		(2294.757)	
Harvard SS^2	.	.	0.003
			(0.006)
N	478	478	478
RMSE	5103.023	5109.889	5107.336
R^2	0.271	0.269	0.272
adj R^2	0.262	0.26	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%) 22333.12     S
N (30%) 26151.76     N
H (30%) 21113.96     H

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
S (30%) 22333.12      S
N (30%) 26151.76      N
H (30%) 21113.96      H

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

Table 5: Categorical Regressions: Student-13

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21113.964*	22333.121*	-1644.587	-45.898
	(423.568)	(404.304)	(2908.426)	(2909.734)
Major: Soc.	1219.157*	.	1598.689*	.
	(585.552)		(556.034)	
Major: Nat.	5037.798*	.	5154.568*	.
	(585.552)		(554.871)	
Major 2: Hum.	.	-1219.157*	.	-1598.689*
		(585.552)		(556.034)
Major 2: Nat.	.	3818.641*	.	3555.879*
		(571.773)		(539.221)
SAT	.	.	10.088*	10.088*
			(1.597)	(1.597)
ACT	.	.	191.452*	191.452*
			(49.55)	(49.55)
Iowa BS	.	.	20.802	20.802
			(26.003)	(26.003)
Prof. Parents: Yes	.	.	802.741	802.741
			(494.745)	(494.745)
Parent Network: Yes	.	.	509.602	509.602
			(472.524)	(472.524)
Gender: Male	.	.	-586.754	-586.754
			(448.043)	(448.043)
N	524	524	495	495
RMSE	5424.311	5424.311	4959.472	4959.472
R^2	0.136	0.136	0.286	0.286
adj R^2	0.133	0.133	0.274	0.274

$*p \leq 0.05$