Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 4
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%	8.17	1172.00	70.18	576.00	2528.00	151000.00	1144.00
25%	19.10	1518.00	92.81	16820.00	19640.00	161700.00	1493.00
50%	22.30	1631.00	99.47	20570.00	23240.00	165500.00	1605.00
75%	25.60	1755.00	106.70	23990.00	27290.00	169300.00	1724.00
100%	36.65	2109.00	131.80	38880.00	38880.00	180900.00	2091.00
mean	22.29	1634.00	99.61	20310.00	23280.00	165400.00	1605.00
sd	4.82	166.30	9.98	5389.00	5694.00	5573.00	164.70
var	23.20	27670.00	99.65	29050000.00	32420000.00	31060000.00	27130.00
NA's	20.00	59.00	0.00	10.00	0.00	0.00	22.00
N	549.00	549.00	549.00	549.00	549.00	549.00	549.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                   major                    pnet
M             :291.0000   N             :187.0000   NO            :383.0000
F             :258.0000   H             :183.0000   YES           :166.0000
NA's          :  0.0000   S             :179.0000   NA's          :  0.0000
entropy       :  0.9974   NA's          :  0.0000   entropy       :  0.8842
normedEntropy:  0.9974   entropy       :  1.5847   normedEntropy:  0.8842
N             :549.0000   normedEntropy:  0.9999   N             :549.0000
                          N             :549.0000
          pprof
NO            :378.0000
YES           :171.0000
NA's          :  0.0000
entropy       :  0.8949
normedEntropy:  0.8949
N             :549.0000
```

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: 0x1522ab8>
act ~ sat + ibs + harv
<environment: 0x1522ab8>
ibs ~ sat + act + harv
<environment: 0x1522ab8>
harv ~ sat + act + ibs
<environment: 0x1522ab8>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998326 0.8546746 0.2015248 0.9998371
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
5972.124574    6.881110    1.252387 6137.258681
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.45 0.37 1.00
act  0.45 1.00 0.39 0.48
ibs  0.37 0.39 1.00 0.37
harv 1.00 0.48 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-4

	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)	2708.163	12597.173*	10877.556*	2360.083	3857.745	3116.966
	(2226.817)	(1071.152)	(2280.777)	(2276.12)	(2908.75)	(2710.821)
SAT	10.926*	.	.	.	56.57	7.999*
	(1.382)				(114.247)	(1.607)
ACT	.	345.507*	.	.	247.726	238.823*
		(46.895)			(130.614)	(54.531)
Iowa BS	.	.	94.721*	.	-6.652	-10.554
			(22.785)		(26.616)	(25.374)
Harvard SS	.	.	.	11.046*	-48.691	.
				(1.386)	(114.218)	
N	517	519	539	481	443	497
RMSE	5118.286	5152.685	5309.673	5002.439	5030.988	5059.854
R^2	0.108	0.095	0.031	0.117	0.124	0.14
adj R^2	0.107	0.093	0.029	0.115	0.116	0.134

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    440 1.1092e+10
2    438 1.1086e+10  2   5817456 0.1149 0.8915
```

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)	2881.606	12438.909*	11871.415*	3397.428	3857.745	3116.966
	(2285.795)	(1105.719)	(2391.011)	(2412.086)	(2908.75)	(2710.821)
SAT	10.817*	.	.	.	56.57	7.999*
	(1.418)				(114.247)	(1.607)
ACT	.	348.805*	.	.	247.726	238.823*
		(48.429)			(130.614)	(54.531)
Iowa BS	.	.	84.116*	.	-6.652	-10.554
			(23.953)		(26.616)	(25.374)
Harvard SS	.	.	.	10.381*	-48.691	.
				(1.471)	(114.218)	
N	497	497	497	443	443	497
RMSE	5149.887	5179.411	5377.477	5077.296	5030.988	5059.854
R^2	0.105	0.095	0.024	0.101	0.124	0.14
adj R^2	0.103	0.093	0.022	0.099	0.116	0.134

$*p \leq 0.05$

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   0.21880861
act   0.19351939
ibs  -0.01872913
```

```
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 0.0432622993
act 0.0334733613
ibs 0.0003018988
```

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%    38.83   36.13   37.27
50%    49.61   47.29   49.18
75%    61.24   59.26   61.80
100%  100.00  100.00  100.00
```

```
mean    49.69    47.31    49.04
sd      16.86    16.37    17.39
var    284.30   268.00   302.50
NA's     0.00     0.00     0.00
N      497.00   497.00   497.00

$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
-16145.7  -3538.0    339.8   3400.3  16850.5

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 13475.793    874.547  15.409  < 2e-16 ***
satpoms        74.979     15.059   4.979 8.85e-07 ***
actpoms        68.017     15.530   4.380 1.45e-05 ***
ibspoms        -6.499     15.625  -0.416    0.678
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5060 on 493 degrees of freedom
Multiple R^2: 0.1397,   Adjusted R^2: 0.1344
F-statistic: 26.67 on 3 and 493 DF,  p-value: 5.296e-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   0.02365285
act   0.09025465
ibs  -0.01194021
harv -0.02036527
```

```
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  0.0004905057
act  0.0071965685
ibs  0.0001249452
harv 0.0003635755
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-4

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	8290.531*	4807.47
	(2875.161)	(2769.327)
SAT	7.346*	7.734*
	(1.685)	(1.576)
ACT	196.47*	236.269*
	(57.904)	(54.222)
Iowa BS	-12.922	-15.381
	(27.222)	(25.661)
Major: Soc.	.	830.598
		(558.709)
Major: Nat.	.	4390.465*
		(553.859)
Prof. Parents: Yes	.	1075.421*
		(490.678)
Parent Network: Yes	.	850.031
		(490.918)
Gender: Male	.	-228.063
		(451.315)
N	507	507
RMSE	5435.039	5060.952
R^2	0.096	0.224
adj R^2	0.091	0.212

$*p \leq 0.05$

```
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 =  225.389815572778 Denominator =  699.725389896758"
```

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

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

```
tval
```

```
difference
 0.3221118
```

```
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
 0.7475032
```

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
225.3898156	699.7253899	0.3221118	498.0000000	0.7475032

```
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    503 14858441457
2    498 12755390049  5 2103051408 16.422 5.156e-15 ***
---
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

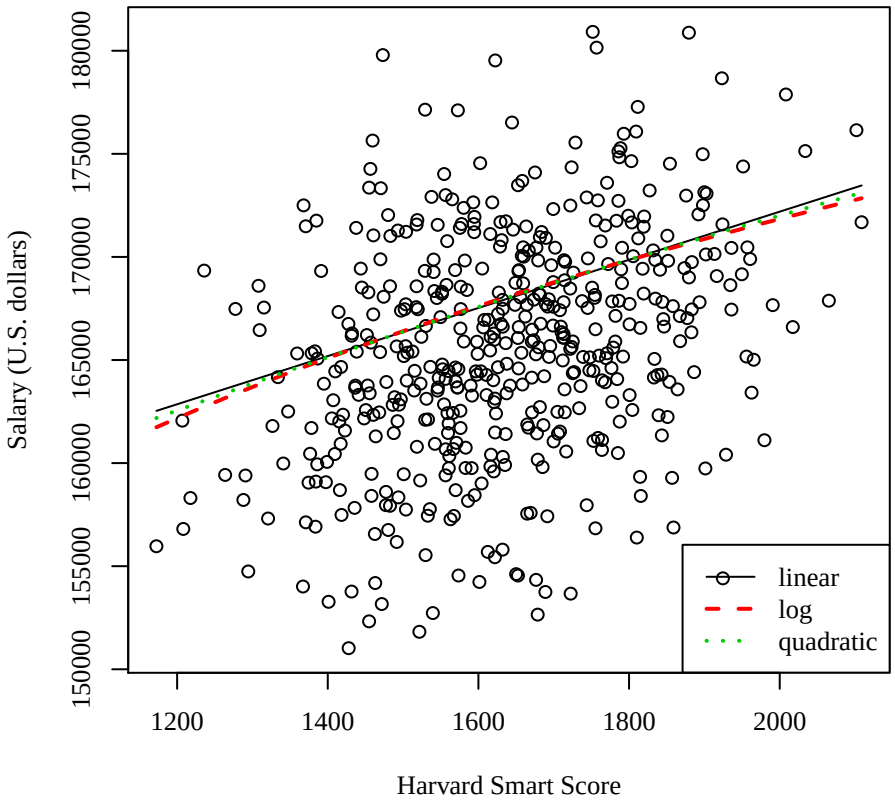
Table 4: Regression with sal3: Student-4

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	144578.22*	23781.483	139709.673*
	(2219.027)	(15681.617)	(15974.489)
Harvard SS	11.676*	.	17.682
	(1.307)		(19.557)
Gender: Male	-582.796	-590.697	-586.334
	(434.205)	(434.146)	(434.765)
Major: Soc.	2057.437*	2064.053*	2059.532*
	(534.883)	(534.828)	(535.428)
Major: Nat.	4846.948*	4851.166*	4849.246*
	(526.151)	(526.117)	(526.697)
Prof. Parents: Yes	466.962	470.936	468.49
	(470.345)	(470.318)	(470.813)
Parent Network: Yes	-679.744	-686.638	-682.81
	(472.289)	(472.226)	(472.837)
ln(Harvard SS)	.	18918.846*	.
		(2116.218)	
Harvard SS^2	.	.	-0.002
			(0.006)
N	490	490	490
RMSE	4786.017	4785.447	4790.509
R^2	0.249	0.249	0.249
adj R^2	0.24	0.24	0.238

$*p \leq 0.05$

```
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
N (30%) 25770.82     N
H (30%) 21601.51     H
S (30%) 22380.93     S

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
N (30%) 25770.82      N
H (30%) 21601.51      H
S (30%) 22380.93      S

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

Table 5: Categorical Regressions: Student-4

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21601.512* (399.492)	22380.934* (403.931)	4807.47 (2769.327)	5638.067* (2713.349)
Major: Soc.	779.422 (568.114)	.	830.598 (558.709)	.
Major: Nat.	4169.308* (561.937)	.	4390.465* (553.859)	.
Major 2: Hum.	.	-779.422 (568.114)	.	-830.598 (558.709)
Major 2: Nat.	.	3389.886* (565.102)	.	3559.867* (555.556)
SAT	.	.	7.734* (1.576)	7.734* (1.576)
ACT	.	.	236.269* (54.222)	236.269* (54.222)
Iowa BS	.	.	-15.381 (25.661)	-15.381 (25.661)
Prof. Parents: Yes	.	.	1075.421* (490.678)	1075.421* (490.678)
Parent Network: Yes	.	.	850.031 (490.918)	850.031 (490.918)
Gender: Male	.	.	-228.063 (451.315)	-228.063 (451.315)
N	549	549	507	507
RMSE	5404.224	5404.224	5060.952	5060.952
R^2	0.102	0.102	0.224	0.224
adj R^2	0.099	0.099	0.212	0.212

$*p \leq 0.05$