Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 39
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%	7.37	1151.00	73.36	6124.00	7775.00	144400.00	1088.00
25%	18.40	1510.00	93.54	17030.00	19690.00	161500.00	1475.00
50%	22.01	1614.00	100.50	21200.00	23870.00	165500.00	1587.00
75%	25.20	1726.00	107.00	24390.00	27450.00	169300.00	1700.00
100%	40.58	2143.00	135.30	35740.00	39500.00	183400.00	2112.00
mean	21.88	1620.00	100.40	20930.00	23780.00	165500.00	1593.00
sd	5.02	164.80	9.96	5469.00	5862.00	6012.00	166.00
var	25.21	27160.00	99.19	29910000.00	34360000.00	36140000.00	27540.00
NA's	23.00	54.00	0.00	17.00	0.00	0.00	26.00
N	526.00	526.00	526.00	526.00	526.00	526.00	526.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                  major                    pnet
M             :276.0000   H             :183.0000   NO            :352.0000
F             :250.0000   N             :175.0000   YES           :174.0000
NA's          :  0.0000   S             :168.0000   NA's          :  0.0000
entropy       :  0.9982   NA's          :  0.0000   entropy       :  0.9157
normedEntropy:  0.9982   entropy       :  1.5841   normedEntropy:  0.9157
N             :526.0000   normedEntropy:  0.9994   N             :526.0000
                          N             :526.0000
          pprof
NO            :369.0000
YES           :157.0000
NA's          :  0.0000
entropy       :  0.8794
normedEntropy:  0.8794
N             :526.0000
```

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: 0x1ecd0d0>
act ~ sat + ibs + harv
<environment: 0x1ecd0d0>
ibs ~ sat + act + harv
<environment: 0x1ecd0d0>
harv ~ sat + act + ibs
<environment: 0x1ecd0d0>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998492 0.8580286 0.2146293 0.9998529
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
6629.775875    7.043674    1.273284 6798.124164
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.42 0.40 1.00
act  0.42 1.00 0.38 0.45
ibs  0.40 0.38 1.00 0.41
harv 1.00 0.45 0.41 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-39

	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)	1554.731	11699.212*	9483.798*	2130.488	203.224	720.494
	(2225.091)	(1041.766)	(2391.651)	(2326.49)	(2886.962)	(2747.571)
SAT	12.217*	.	.	.	18.535	8.602*
	(1.389)				(118.661)	(1.596)
ACT	.	421.958*	.	.	327.561*	307.751*
		(46.485)			(129.124)	(53.201)
Iowa BS	.	.	114.007*	.	6.373	-1.64
			(23.705)		(27.474)	(26.259)
Harvard SS	.	.	.	11.596*	-10.313	.
				(1.428)	(118.609)	
N	483	487	509	456	415	461
RMSE	5088.949	5096.656	5353.989	5053.749	4903.632	4944.6
R^2	0.139	0.145	0.044	0.127	0.209	0.202
adj R^2	0.137	0.143	0.042	0.125	0.201	0.197

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    412 9860207720
2    410 9858697763  2   1509957 0.0314 0.9691
```

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)	1213.717	11633.492*	9469.179*	785.367	203.224	720.494
	(2306.294)	(1080.068)	(2557.737)	(2425.877)	(2886.962)	(2747.571)
SAT	12.426*	.	.	.	18.535	8.602*
	(1.439)				(118.661)	(1.596)
ACT	.	427.77*	.	.	327.561*	307.751*
		(48.02)			(129.124)	(53.201)
Iowa BS	.	.	115.187*	.	6.373	-1.64
			(25.384)		(27.474)	(26.259)
Harvard SS	.	.	.	12.465*	-10.313	.
				(1.49)	(118.609)	
N	461	461	461	415	415	461
RMSE	5123.237	5100.304	5403.738	5079.16	4903.632	4944.6
R^2	0.14	0.147	0.043	0.145	0.209	0.202
adj R^2	0.138	0.146	0.041	0.143	0.201	0.197

$*p \leq 0.05$

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 -1.000000000
sat   0.244490072
act   0.261200966
ibs  -0.002921458
```

```
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 5.072353e-02
act 5.841947e-02
ibs 6.809601e-06
```

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%    33.48   32.69   37.85
50%    44.17   43.13   48.90
75%    53.69   54.21   59.67
100%  100.00  100.00  100.00
```

```
mean    43.88    43.47    49.39
sd      14.91    16.03    16.20
var    222.40   257.00   262.30
NA's     0.00     0.00     0.00
N      461.00   461.00   461.00

$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
-12228.3  -3240.1      2.7   3205.8  14462.8

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 12225.451    894.646  13.665  < 2e-16 ***
satpoms        88.148     16.354   5.390 1.13e-07 ***
actpoms       102.204     17.668   5.785 1.35e-08 ***
ibspoms        -1.015     16.257  -0.062     0.95
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4945 on 457 degrees of freedom
Multiple R^2: 0.2022,   Adjusted R^2: 0.1969
F-statistic:  38.6 on 3 and 457 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   0.007714101
act   0.124311117
ibs   0.011454707
harv -0.004294070
```

```
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   0.00004708753
act   0.01241918211
ibs   0.00010383274
harv  0.00001459001
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-39

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	4759.143	1416.791
	(2974.969)	(2824.657)
SAT	7.682*	8.033*
	(1.722)	(1.596)
ACT	304.11*	282.299*
	(56.614)	(52.585)
Iowa BS	2.295	8.083
	(28.453)	(26.695)
Major: Soc.	.	2131.588*
		(565.907)
Major: Nat.	.	4698.851*
		(561.748)
Prof. Parents: Yes	.	1544.145*
		(497.03)
Parent Network: Yes	.	661.772
		(493.061)
Gender: Male	.	-538.809
		(460.537)
N	477	477
RMSE	5413.111	5001.304
R^2	0.161	0.291
adj R^2	0.155	0.279

$*p \leq 0.05$

```
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 =  882.372737924557 Denominator =  710.932478384869"
```

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

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

```
tval
```

```
difference
  1.241148
```

```
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
 0.2151727
```

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
     882.3727379       710.9324784         1.2411484       468.0000000         0.2151727
```

```
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    473 13859739595
2    468 11706101593  5 2153638002 17.22 1.185e-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

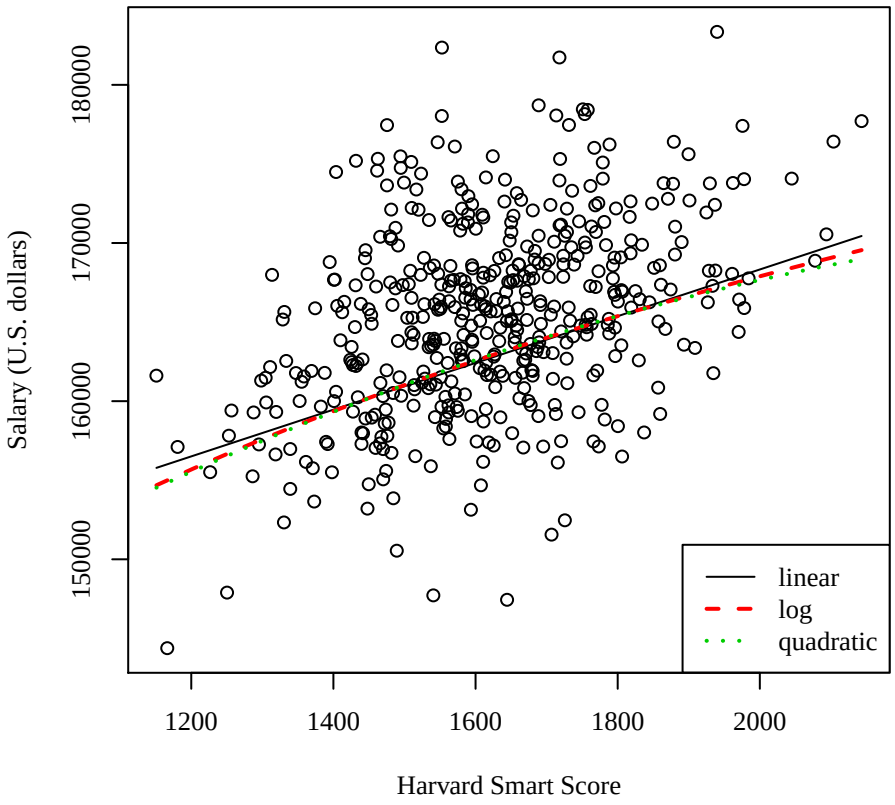
Table 4: Regression with sal3: Student-39

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	138171.325*	-14489.862	121622.932*
	(2312.686)	(16527.129)	(15603.828)
Harvard SS	14.773*	.	35.296
	(1.393)		(19.189)
Gender: Male	609.648	616.846	620.728
	(460.771)	(460.055)	(460.813)
Major: Soc.	2586.384*	2597.677*	2606.169*
	(564.248)	(563.427)	(564.459)
Major: Nat.	5470.682*	5449.499*	5447.613*
	(549.075)	(548.186)	(549.408)
Prof. Parents: Yes	1153.315*	1151.256*	1148.764*
	(501.253)	(500.497)	(501.19)
Parent Network: Yes	106.381	69.398	56.719
	(491.659)	(490.923)	(493.757)
ln(Harvard SS)	.	23914.146*	.
		(2236.885)	
Harvard SS^2	.	.	-0.006
			(0.006)
N	472	472	472
RMSE	4973.082	4965.546	4972.28
R^2	0.317	0.319	0.319
adj R^2	0.308	0.311	0.309

$*p \leq 0.05$

```
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%) 21645.00     H
N (30%) 26243.65     N
S (30%) 23549.25     S

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
H (30%) 21645.00      H
N (30%) 26243.65      N
S (30%) 23549.25      S

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

Table 5: Categorical Regressions: Student-39

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21644.996* (410.559)	23549.248* (428.496)	1416.791 (2824.657)	3548.379 (2769.652)
Major: Soc.	1904.251* (593.437)	.	2131.588* (565.907)	.
Major: Nat.	4598.65* (587.216)	.	4698.851* (561.748)	.
Major 2: Hum.	.	-1904.251* (593.437)	.	-2131.588* (565.907)
Major 2: Nat.	.	2694.398* (599.894)	.	2567.263* (567.96)
SAT	.	.	8.033* (1.596)	8.033* (1.596)
ACT	.	.	282.299* (52.585)	282.299* (52.585)
Iowa BS	.	.	8.083 (26.695)	8.083 (26.695)
Prof. Parents: Yes	.	.	1544.145* (497.03)	1544.145* (497.03)
Parent Network: Yes	.	.	661.772 (493.061)	661.772 (493.061)
Gender: Male	.	.	-538.809 (460.537)	-538.809 (460.537)
N	526	526	477	477
RMSE	5553.94	5553.94	5001.304	5001.304
R^2	0.106	0.106	0.291	0.291
adj R^2	0.102	0.102	0.279	0.279

$*p \leq 0.05$