Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 11
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%	8.07	1179.00	72.64	1060.00	3966.00	147500.00	1160.00
25%	18.53	1500.00	93.28	16660.00	19610.00	161100.00	1479.00
50%	21.54	1612.00	100.60	20490.00	23400.00	165100.00	1591.00
75%	24.76	1719.00	106.60	24340.00	27420.00	169100.00	1698.00
100%	36.47	2104.00	125.10	42300.00	45580.00	179900.00	2283.00
mean	21.65	1614.00	99.92	20360.00	23340.00	165100.00	1594.00
sd	4.83	157.00	9.97	5622.00	6031.00	5655.00	158.40
var	23.33	24650.00	99.46	31610000.00	36370000.00	31980000.00	25080.00
NA's	17.00	57.00	0.00	9.00	0.00	0.00	22.00
N	568.00	568.00	568.00	568.00	568.00	568.00	568.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
           gender                    major                     pnet
F            :288.0000   S            :201.0000   NO           :380.0000
M            :280.0000   H            :186.0000   YES          :188.0000
NA's         :  0.0000   N            :181.0000   NA's         :  0.0000
entropy      :  0.9999   NA's         :  0.0000   entropy      :  0.9159
normedEntropy:  0.9999   entropy      :  1.5835   normedEntropy:  0.9159
N            :568.0000   normedEntropy:  0.9991   N            :568.0000
                         N            :568.0000
           pprof
NO           :377.0000
YES          :191.0000
NA's         :  0.0000
entropy      :  0.9212
normedEntropy:  0.9212
N            :568.0000
```

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: 0x1fbaeb0>
act ~ sat + ibs + harv
<environment: 0x1fbaeb0>
ibs ~ sat + act + harv
<environment: 0x1fbaeb0>
harv ~ sat + act + ibs
<environment: 0x1fbaeb0>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998322 0.8609103 0.2641857 0.9998370
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
5959.787342    7.189603    1.359039 6135.108776
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.44 0.46 1.00
act  0.44 1.00 0.40 0.46
ibs  0.46 0.40 1.00 0.46
harv 1.00 0.46 0.46 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-11

	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)	2117.103	12692.43*	7298.998*	1168.023	161.682	-620.105
	(2345.396)	(1055.885)	(2325.27)	(2479.906)	(2931.304)	(2749.135)
SAT	11.458*	.	.	.	22.953	8.199*
	(1.464)				(122.024)	(1.71)
ACT	.	353.136*	.	.	240.452	206.905*
		(47.614)			(133.381)	(54.045)
Iowa BS	.	.	130.791*	.	21.899	34.398
			(23.175)		(28.167)	(26.625)
Harvard SS	.	.	.	11.905*	-14.686	.
				(1.529)	(122.092)	
N	537	542	559	503	469	521
RMSE	5351.549	5377.204	5472.714	5355.287	5298.979	5257.048
R^2	0.103	0.092	0.054	0.108	0.143	0.142
adj R^2	0.101	0.091	0.052	0.106	0.135	0.137

$*p \le 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    466 1.3046e+10
2    464 1.3029e+10  2  17101668 0.3045 0.7376
```

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)	1480.425	12883.29*	7125.525*	872.592	161.682	-620.105
	(2388.116)	(1080.041)	(2412.691)	(2568.356)	(2931.304)	(2749.135)
SAT	11.846*	.	.	.	22.953	8.199*
	(1.491)				(122.024)	(1.71)
ACT	.	345.863*	.	.	240.452	206.905*
		(48.703)			(133.381)	(54.045)
Iowa BS	.	.	132.469*	.	21.899	34.398
			(24.017)		(28.167)	(26.625)
Harvard SS	.	.	.	12.089*	-14.686	.
				(1.582)	(122.092)	
N	521	521	521	469	469	521
RMSE	5349.801	5409.215	5506.83	5378.891	5298.979	5257.048
R^2	0.108	0.089	0.055	0.111	0.143	0.142
adj R^2	0.107	0.087	0.054	0.109	0.135	0.137

$*p \leq 0.05$

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   0.20633064
act   0.16603362
ibs   0.05672786
```

```
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  0.038131749
act  0.024310711
ibs  0.002768585
```

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%    36.27   39.58   28.54
50%    47.32   53.14   38.47
75%    58.87   65.34   47.89
100%  100.00  100.00  100.00
```

```
mean    47.77    52.05    38.64
sd      17.15    19.16    14.02
var    294.10   367.10   196.40
NA's     0.00     0.00     0.00
N      521.00   521.00   521.00

$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
-15392.5  -3546.0    147.9   3475.7  22396.8

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 13062.46     841.60  15.521  < 2e-16 ***
satpoms        92.06      19.20   4.795 2.13e-06 ***
actpoms        58.76      15.35   3.828 0.000145 ***
ibspoms        18.05      13.97   1.292 0.196956
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5257 on 517 degrees of freedom
Multiple R^2: 0.1424,  Adjusted R^2: 0.1375
F-statistic: 28.62 on 3 and 517 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   0.008731947
act   0.083399096
ibs   0.036070138
harv -0.005584101
```

```
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  0.00006535807
act  0.00600340809
ibs  0.00111661882
harv 0.00002672784
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-11

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	2677.224	-1288.028
	(2941.474)	(2795.005)
SAT	9.137*	8.233*
	(1.826)	(1.698)
ACT	212.158*	203.948*
	(58.202)	(54.034)
Iowa BS	15.497	37.15
	(28.646)	(26.64)
Major: Soc.	.	2689.822*
		(558.367)
Major: Nat.	.	5332.506*
		(574.55)
Prof. Parents: Yes	.	795.616
		(488.681)
Parent Network: Yes	.	1002.33*
		(488.594)
Gender: Male	.	263.794
		(459.374)
N	530	530
RMSE	5679.503	5258.122
R^2	0.128	0.26
adj R^2	0.123	0.249
$*p \leq 0.05$		

```
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 =  -206.714410153269 Denominator =  696.104327403967"
```

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

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

```
tval
```

```
difference
-0.2969589
```

```
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
 0.7666162
```

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
     -206.7144102       696.1043274        -0.2969589       521.0000000         0.7666162
```

```
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    526 16967055290
2    521 14404529408  5 2562525881 18.537 < 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

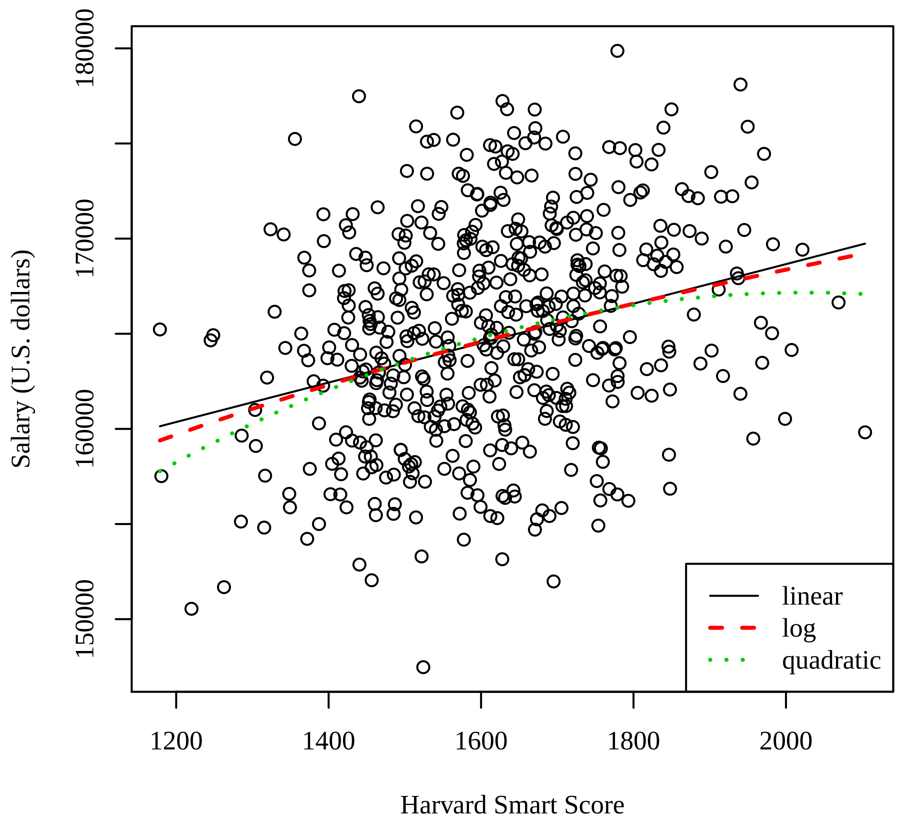
Table 4: Regression with sal3: Student-11

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	145122.583*	36533.812*	110637.211*
	(2337.283)	(16805.715)	(17253.018)
Harvard SS	10.373*	.	53.035*
	(1.415)		(21.195)
Gender: Male	475.654	480.132	491.967
	(443.154)	(442.396)	(441.885)
Major: Soc.	2790.391*	2798.182*	2822.233*
	(535.808)	(534.875)	(534.417)
Major: Nat.	5264.283*	5270.122*	5284.033*
	(556.793)	(555.831)	(555.192)
Prof. Parents: Yes	1648.869*	1660.096*	1678.949*
	(471.101)	(470.348)	(469.909)
Parent Network: Yes	-479.532	-467.572	-416.891
	(470.355)	(469.386)	(469.956)
ln(Harvard SS)	.	16976.545*	.
		(2275.584)	
Harvard SS^2	.	.	-0.013*
			(0.006)
N	511	511	511
RMSE	4995.856	4987.227	4980.712
R^2	0.229	0.232	0.235
adj R^2	0.22	0.223	0.225

$*p \leq 0.05$

```
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 (40%) 23366.01     S
H (30%) 20714.12     H
N (30%) 26006.09     N

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

```
predictOMatic(cm2)
```

```
$major2
              fit  major2
S (40%) 23366.01      S
H (30%) 20714.12      H
N (30%) 26006.09      N

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

Table 5: Categorical Regressions: Student-11

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	20714.117*	23366.012*	-1288.028	1401.794
	(414.497)	(398.731)	(2795.005)	(2775.521)
Major: Soc.	2651.895*	.	2689.822*	.
	(575.147)		(558.367)	
Major: Nat.	5291.973*	.	5332.506*	.
	(590.222)		(574.55)	
Major 2: Hum.	.	-2651.895*	.	-2689.822*
		(575.147)		(558.367)
Major 2: Nat.	.	2640.078*	.	2642.684*
		(579.258)		(558.573)
SAT	.	.	8.233*	8.233*
			(1.698)	(1.698)
ACT	.	.	203.948*	203.948*
			(54.034)	(54.034)
Iowa BS	.	.	37.15	37.15
			(26.64)	(26.64)
Prof. Parents: Yes	.	.	795.616	795.616
			(488.681)	(488.681)
Parent Network: Yes	.	.	1002.33*	1002.33*
			(488.594)	(488.594)
Gender: Male	.	.	263.794	263.794
			(459.374)	(459.374)
N	568	568	530	530
RMSE	5652.984	5652.984	5258.122	5258.122
R^2	0.125	0.125	0.26	0.26
adj R^2	0.121	0.121	0.249	0.249

$*p \leq 0.05$