Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 24
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	1173.00	69.56	5431.00	7497.00	149900.00	1162.00
25%	18.88	1521.00	94.19	17180.00	19750.00	161600.00	1504.00
50%	22.10	1626.00	100.70	20540.00	23540.00	165300.00	1605.00
75%	25.27	1731.00	107.90	24250.00	27670.00	169500.00	1708.00
100%	35.76	2049.00	132.00	35910.00	40900.00	186000.00	2148.00
mean	22.14	1627.00	100.80	20620.00	23650.00	165400.00	1607.00
sd	5.00	156.30	10.35	5398.00	5857.00	5811.00	154.70
var	24.97	24430.00	107.10	29130000.00	34300000.00	33760000.00	23920.00
NA's	10.00	62.00	0.00	10.00	0.00	0.00	25.00
N	529.00	529.00	529.00	529.00	529.00	529.00	529.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                   major                    pnet
M             :284.0000   N             :182.0000   NO            :358.0000
F             :245.0000   S             :177.0000   YES           :171.0000
NA's          :  0.0000   H             :170.0000   NA's          :  0.0000
entropy       :  0.9961   NA's          :  0.0000   entropy       :  0.9079
normedEntropy:  0.9961   entropy       :  1.5844   normedEntropy:  0.9079
N             :529.0000   normedEntropy:  0.9996   N             :529.0000
                          N             :529.0000
          pprof
NO            :375.0000
YES           :154.0000
NA's          :  0.0000
entropy       :  0.8702
normedEntropy:  0.8702
N             :529.0000
```

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: 0x1c6ec70>
act ~ sat + ibs + harv
<environment: 0x1c6ec70>
ibs ~ sat + act + harv
<environment: 0x1c6ec70>
harv ~ sat + act + ibs
<environment: 0x1c6ec70>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998188 0.8649658 0.2728087 0.9998239
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
5519.161404    7.405529    1.375154 5678.197193
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.42 0.38 1.00
act  0.42 1.00 0.49 0.45
ibs  0.38 0.49 1.00 0.39
harv 1.00 0.45 0.39 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-24

	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)	4492.215	14507.903*	12360.976*	5746.479*	6583.15*	4413.859
	(2439.836)	(1049.019)	(2286.498)	(2519.023)	(3044.577)	(2843.66)
SAT	9.976*	.	.	.	7.529	8.002*
	(1.511)				(119.871)	(1.686)
ACT	.	278.377*	.	.	161.655	173.145*
		(46.239)			(135.625)	(55.466)
Iowa BS	.	.	81.992*	.	-3.982	-5.234
			(22.576)		(27.14)	(26.182)
Harvard SS	.	.	.	9.056*	-0.886	.
				(1.541)	(119.784)	
N	494	509	519	457	428	484
RMSE	5162.838	5210.874	5335.177	5116.027	5049.715	5098.411
R^2	0.081	0.067	0.025	0.071	0.082	0.105
adj R^2	0.079	0.065	0.023	0.068	0.073	0.099

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    425 1.0787e+10
2    423 1.0786e+10  2    550047 0.0108 0.9893
```

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)	4376.381	14641.817*	12425.802*	6177.079*	6583.15*	4413.859
	(2438.968)	(1085.449)	(2344.686)	(2607.124)	(3044.577)	(2843.66)
SAT	10.08*	.	.	.	7.529	8.002*
	(1.511)				(119.871)	(1.686)
ACT	.	268.156*	.	.	161.655	173.145*
		(47.894)			(135.625)	(55.466)
Iowa BS	.	.	80.854*	.	-3.982	-5.234
			(23.146)		(27.14)	(26.182)
Harvard SS	.	.	.	8.741*	-0.886	.
				(1.598)	(119.784)	
N	484	484	484	428	428	484
RMSE	5145.477	5211.011	5310.99	5075.234	5049.715	5098.411
R^2	0.085	0.061	0.025	0.066	0.082	0.105
adj R^2	0.083	0.059	0.023	0.063	0.073	0.099

$*p \leq 0.05$

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   0.211727076
act   0.141057948
ibs  -0.009123892
```

```
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 4.200747e-02
act 1.817100e-02
ibs 7.451635e-05
```

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%    40.62   39.50   34.57
50%    51.69   49.78   44.83
75%    63.02   61.05   55.76
100%  100.00  100.00  100.00
```

```
mean    51.95    49.94    45.10
sd      17.44    16.71    15.70
var    304.10   279.30   246.60
NA's     0.00     0.00     0.00
N      484.00   484.00   484.00

$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
-16935.7  -3546.1    -87.1   3417.0  14363.2

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 14621.44     900.85  16.231  < 2e-16 ***
satpoms        78.95      16.64   4.746 2.74e-06 ***
actpoms        49.16      15.75   3.122  0.00191 **
ibspoms        -3.27      16.36  -0.200  0.84164
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5098 on 480 degrees of freedom
Multiple R^2: 0.1049,   Adjusted R^2: 0.09934
F-statistic: 18.76 on 3 and 480 DF,  p-value: 1.609e-11
```

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 -1.0000000000
sat   0.0030540009
act   0.0578564476
ibs  -0.0071338798
harv -0.0003597392
```

```
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  0.0000085664516
act  0.0030847382467
ibs  0.0000467446900
harv 0.0000001188596
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-24

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	7534.377*	4583.263
	(3089.599)	(2872.694)
SAT	6.489*	7.761*
	(1.828)	(1.697)
ACT	189.133*	174.525*
	(60.368)	(55.638)
Iowa BS	14.42	-8.524
	(28.556)	(26.236)
Major: Soc.	.	2541.799*
		(575.727)
Major: Nat.	.	5431.36*
		(567.232)
Prof. Parents: Yes	.	1017.252*
		(513.005)
Parent Network: Yes	.	1419.593*
		(498.326)
Gender: Male	.	150.675
		(464.422)
N	494	494
RMSE	5604.533	5113.957
R^2	0.086	0.247
adj R^2	0.08	0.234

$*p \leq 0.05$

```
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 =  -402.341027806567 Denominator =  698.867120325514"
```

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

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

```
tval
```

```
difference
-0.5757046
```

```
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
 0.5650819
```

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
    -402.3410278       698.8671203        -0.5757046       485.0000000         0.5650819
```

```
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    490 15391286009
2    485 12683991398  5 2707294611 20.704 < 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

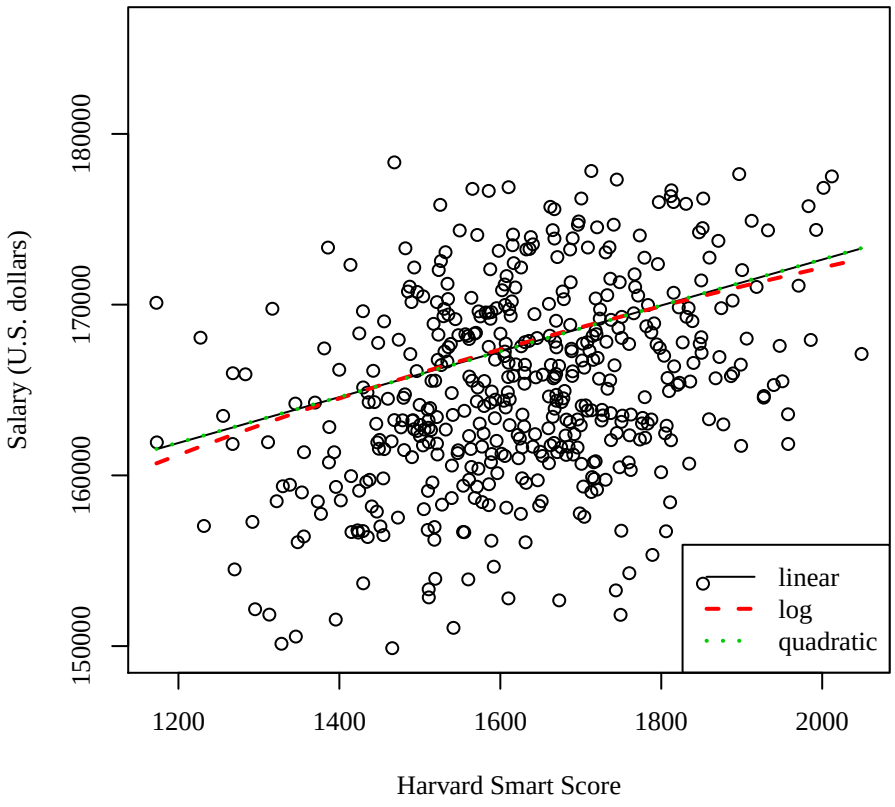
Table 4: Regression with sal3: Student-24

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	140764.903*	3944.752	141098.85*
	(2394.468)	(17031.149)	(17505.775)
Harvard SS	13.426*	.	13.009
	(1.436)		(21.657)
Gender: Male	-672.125	-665.37	-672.231
	(448.744)	(448.947)	(449.266)
Major: Soc.	1880.342*	1847.464*	1881.304*
	(550.06)	(550.228)	(552.92)
Major: Nat.	5694.577*	5677.829*	5695.086*
	(547.623)	(547.798)	(548.855)
Prof. Parents: Yes	1061.226*	1070.886*	1060.824*
	(500.841)	(501.129)	(501.822)
Parent Network: Yes	195.893	181.313	196.411
	(477.182)	(477.548)	(478.457)
ln(Harvard SS)	.	21472.646*	.
		(2303.484)	
Harvard SS^2	.	.	0
			(0.007)
N	467	467	467
RMSE	4817.82	4820.211	4823.063
R^2	0.299	0.298	0.299
adj R^2	0.289	0.289	0.288

$*p \leq 0.05$

```
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%) 26266.44     N
S (30%) 23518.52     S
H (30%) 20983.21     H

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

```
predictOMatic(cm2)
```

```
$major2
             fit major2
N (30%) 26266.44      N
S (30%) 23518.52      S
H (30%) 20983.21      H

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

Table 5: Categorical Regressions: Student-24

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	20983.208* (418.416)	23518.52* (410.059)	4583.263 (2872.694)	7125.062* (2848.881)
Major: Soc.	2535.312* (585.851)	.	2541.799* (575.727)	.
Major: Nat.	5283.231* (581.895)	.	5431.36* (567.232)	.
Major 2: Hum.	.	-2535.312* (585.851)	.	-2541.799* (575.727)
Major 2: Nat.	.	2747.918* (575.915)	.	2889.561* (562.886)
SAT	.	.	7.761* (1.697)	7.761* (1.697)
ACT	.	.	174.525* (55.638)	174.525* (55.638)
Iowa BS	.	.	-8.524 (26.236)	-8.524 (26.236)
Prof. Parents: Yes	.	.	1017.252* (513.005)	1017.252* (513.005)
Parent Network: Yes	.	.	1419.593* (498.326)	1419.593* (498.326)
Gender: Male	.	.	150.675 (464.422)	150.675 (464.422)
N	529	529	494	494
RMSE	5455.483	5455.483	5113.957	5113.957
R^2	0.136	0.136	0.247	0.247
adj R^2	0.132	0.132	0.234	0.234

$*p \leq 0.05$