Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 42
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%	6.87	1097.00	68.12	6165.00	6749.00	149000.00	1089.00
25%	18.97	1511.00	93.57	16960.00	19660.00	162000.00	1491.00
50%	21.73	1617.00	100.30	20460.00	23380.00	165700.00	1592.00
75%	25.42	1743.00	106.30	23750.00	26230.00	169500.00	1711.00
100%	34.95	2084.00	133.90	35750.00	39550.00	179900.00	2048.00
mean	22.07	1627.00	99.92	20370.00	23220.00	165700.00	1603.00
sd	4.72	165.80	9.94	5193.00	5637.00	5550.00	162.80
var	22.29	27480.00	98.86	26960000.00	31770000.00	30810000.00	26500.00
NA's	14.00	69.00	0.00	10.00	0.00	0.00	29.00
N	535.00	535.00	535.00	535.00	535.00	535.00	535.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                   major                    pnet
F             :273.0000   H             :191.0000   NO            :378.0000
M             :262.0000   N             :175.0000   YES           :157.0000
NA's          :  0.0000   S             :169.0000   NA's          :  0.0000
entropy       :  0.9997   NA's          :  0.0000   entropy       :  0.8731
normedEntropy:  0.9997   entropy       :  1.5830   normedEntropy:  0.8731
N             :535.0000   normedEntropy:  0.9988   N             :535.0000
                          N             :535.0000
          pprof
NO            :381.0000
YES           :154.0000
NA's          :  0.0000
entropy       :  0.8659
normedEntropy:  0.8659
N             :535.0000
```

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: 0x3040f50>
act ~ sat + ibs + harv
<environment: 0x3040f50>
ibs ~ sat + act + harv
<environment: 0x3040f50>
harv ~ sat + act + ibs
<environment: 0x3040f50>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998517 0.8612414 0.2096369 0.9998552
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
6743.133227    7.206760    1.265241 6905.568609
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.40 0.37 1.00
act  0.40 1.00 0.39 0.43
ibs  0.37 0.39 1.00 0.37
harv 1.00 0.43 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-42

	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)	1736.154	13821.081*	13773.591*	1814.057	1807.734	2426.657
	(2140.099)	(1055.525)	(2257.754)	(2242.37)	(2755.233)	(2624.834)
SAT	11.546*	.	.	.	135.085	10.19*
	(1.329)				(113.82)	(1.5)
ACT	.	296.409*	.	.	325.431*	179.649*
		(46.872)			(128.704)	(51.826)
Iowa BS	.	.	66.068*	.	-30.188	-25.124
			(22.492)		(25.678)	(24.553)
Harvard SS	.	.	.	11.412*	-124.551	.
				(1.373)	(113.752)	
N	497	511	525	456	419	483
RMSE	4799.458	5003.098	5155.358	4830.683	4664.227	4730.398
R^2	0.132	0.073	0.016	0.132	0.18	0.159
adj R^2	0.131	0.071	0.014	0.13	0.172	0.154

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    416 9059693520
2    414 9006576020  2  53117500 1.2208 0.2961
```

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)	1373.533	13547.191*	13037.461*	462.398	1807.734	2426.657
	(2155.161)	(1072.185)	(2368.235)	(2267.912)	(2755.233)	(2624.834)
SAT	11.755*	.	.	.	135.085	10.19*
	(1.339)				(113.82)	(1.5)
ACT	.	301.177*	.	.	325.431*	179.649*
		(47.446)			(128.704)	(51.826)
Iowa BS	.	.	71.554*	.	-30.188	-25.124
			(23.54)		(25.678)	(24.553)
Harvard SS	.	.	.	12.091*	-124.551	.
				(1.385)	(113.752)	
N	483	483	483	419	419	483
RMSE	4779.401	4945.282	5099.505	4719.605	4664.227	4730.398
R^2	0.138	0.077	0.019	0.155	0.18	0.159
adj R^2	0.136	0.075	0.017	0.152	0.172	0.154

$*p \leq 0.05$

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   0.29648431
act   0.15643213
ibs  -0.04670353
```

```
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  0.081026800
act  0.021090070
ibs  0.001837867
```

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%    43.25   35.92   41.47
50%    52.74   47.13   52.29
75%    66.06   56.89   64.71
100%  100.00  100.00  100.00
```

```
mean    54.22    46.32    53.45
sd      16.91    15.67    16.95
var    285.80   245.70   287.20
NA's     0.00     0.00     0.00
N      483.00   483.00   483.00

$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
-14416.4  -3316.2    189.2   3213.5  13921.5

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 12972.99     895.31  14.490  < 2e-16 ***
satpoms        97.77      14.39   6.794 3.24e-11 ***
actpoms        50.45      14.55   3.466 0.000575 ***
ibspoms       -15.82      15.46  -1.023 0.306695
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4730 on 479 degrees of freedom
Multiple R^2: 0.1593,   Adjusted R^2: 0.154
F-statistic: 30.24 on 3 and 479 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   0.05823034
act   0.12332203
ibs  -0.05768351
harv -0.05373523
```

```
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   0.002789304
act   0.012660738
ibs   0.002736988
harv  0.002374084
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-42

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	3481.582	2642.696
	(2872.029)	(2660.953)
SAT	11.533*	10.172*
	(1.63)	(1.498)
ACT	157.855*	180.997*
	(56.596)	(51.901)
Iowa BS	-23.972	-30.136
	(26.967)	(24.779)
Major: Soc.	.	1561.413*
		(526.465)
Major: Nat.	.	4812.727*
		(516.985)
Prof. Parents: Yes	.	1443.094*
		(483.312)
Parent Network: Yes	.	843.479
		(474.856)
Gender: Male	.	774.118
		(431.554)
N	492	492
RMSE	5211.319	4748.576
R^2	0.152	0.303
adj R^2	0.147	0.292

$*p \leq 0.05$

```
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 =  599.614860983146 Denominator =  664.374824470839"
```

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

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

```
tval
```

```
difference
  0.902525
```

```
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
 0.3672279
```

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
     599.6148610       664.3748245         0.9025250       483.0000000         0.3672279
```

```
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    488 13253028260
2    483 10891152503  5 2361875757 20.949 < 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

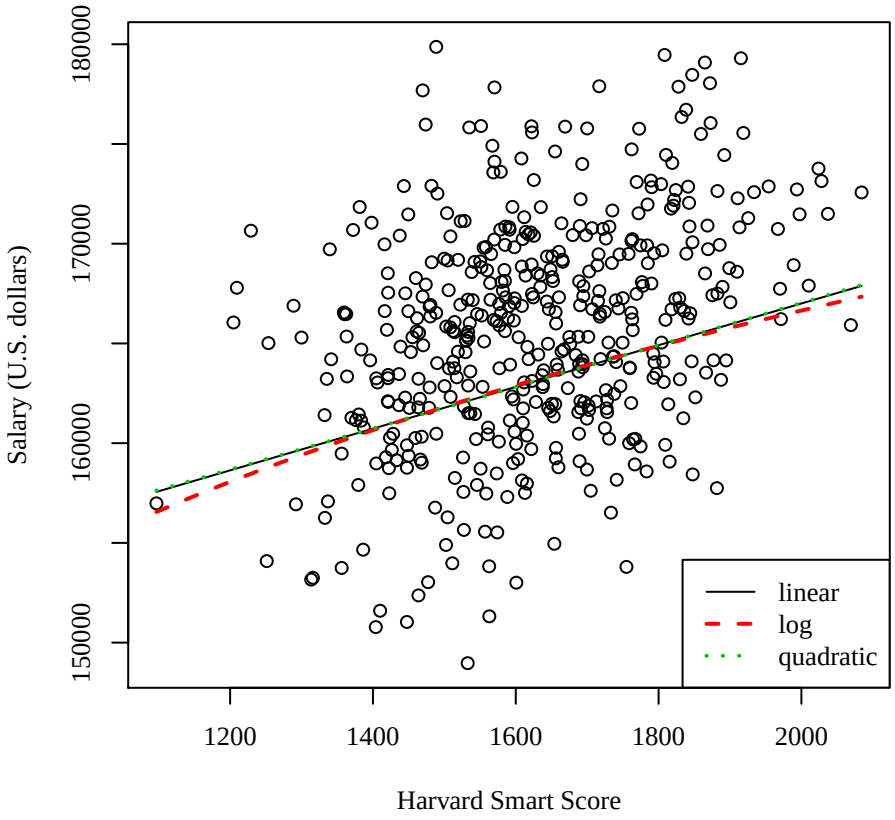
Table 4: Regression with sal3: Student-42

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	146086.166*	38890.281*	146763.823*
	(2160.287)	(15600.872)	(15779.782)
Harvard SS	10.459*	.	9.621
	(1.31)		(19.381)
Gender: Male	-809.848	-811.545	-809.894
	(431.803)	(431.966)	(432.274)
Major: Soc.	2406.531*	2415.161*	2405.969*
	(522.147)	(522.341)	(522.876)
Major: Nat.	5294.015*	5312.731*	5292.499*
	(520.354)	(520.462)	(522.093)
Prof. Parents: Yes	2214.444*	2234.29*	2213.037*
	(476.89)	(476.823)	(478.512)
Parent Network: Yes	-308.703	-320.604	-307.765
	(472.498)	(472.798)	(473.508)
ln(Harvard SS)	.	16808.5*	.
		(2112.68)	
Harvard SS^2	.	.	0
			(0.006)
N	466	466	466
RMSE	4626.383	4628.229	4631.421
R^2	0.315	0.315	0.315
adj R^2	0.306	0.306	0.305

$*p \leq 0.05$

```
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 (40%) 21166.97     H
N (30%) 25918.76     N
S (30%) 22749.12     S

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

```
predictOMatic(cm2)
```

```
$major2
           fit major2
H (40%) 21166.97      H
N (30%) 25918.76      N
S (30%) 22749.12      S

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

Table 5: Categorical Regressions: Student-42

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21166.967* (382.293)	22749.116* (406.415)	2642.696 (2660.953)	4204.11 (2667.893)
Major: Soc.	1582.149* (557.962)	.	1561.413* (526.465)	.
Major: Nat.	4751.796* (552.864)	.	4812.727* (516.985)	.
Major 2: Hum.	.	-1582.149* (557.962)	.	-1561.413* (526.465)
Major 2: Nat.	.	3169.647* (569.81)	.	3251.313* (536.276)
SAT	.	.	10.172* (1.498)	10.172* (1.498)
ACT	.	.	180.997* (51.901)	180.997* (51.901)
Iowa BS	.	.	-30.136 (24.779)	-30.136 (24.779)
Prof. Parents: Yes	.	.	1443.094* (483.312)	1443.094* (483.312)
Parent Network: Yes	.	.	843.479 (474.856)	843.479 (474.856)
Gender: Male	.	.	774.118 (431.554)	774.118 (431.554)
N	535	535	492	492
RMSE	5283.397	5283.397	4748.576	4748.576
R^2	0.125	0.125	0.303	0.303
adj R^2	0.121	0.121	0.292	0.292

$*p \leq 0.05$