Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 14
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%	7.70	1155.00	65.72	1378.00	2397.00	148200.00	1143.00
25%	19.02	1522.00	93.25	16510.00	19330.00	161800.00	1503.00
50%	22.21	1627.00	99.86	20420.00	23210.00	165400.00	1606.00
75%	25.84	1742.00	107.10	24230.00	26980.00	168700.00	1719.00
100%	36.02	2134.00	132.90	35100.00	40700.00	183200.00	2098.00
mean	22.24	1630.00	100.20	20430.00	23240.00	165400.00	1609.00
sd	5.21	167.60	9.98	5410.00	5649.00	5476.00	163.30
var	27.16	28080.00	99.51	29270000.00	31910000.00	29980000.00	26660.00
NA's	12.00	54.00	0.00	13.00	0.00	0.00	28.00
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
F            :267.0000    H            :190.0000    NO           :379.0000
M            :259.0000    S            :172.0000    YES          :147.0000
NA's         :  0.0000    N            :164.0000    NA's         :  0.0000
entropy      :  0.9998    NA's         :  0.0000    entropy      :  0.8547
normedEntropy:  0.9998    entropy      :  1.5822    normedEntropy:  0.8547
N            :526.0000    normedEntropy:  0.9983    N            :526.0000
                          N            :526.0000
            pprof
NO           :364.0000
YES          :162.0000
NA's         :  0.0000
entropy      :  0.8908
normedEntropy:  0.8908
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: 0x25feec8>
act ~ sat + ibs + harv
<environment: 0x25feec8>
ibs ~ sat + act + harv
<environment: 0x25feec8>
harv ~ sat + act + ibs
<environment: 0x25feec8>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998423 0.8666766 0.2242904 0.9998465
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
6341.554378    7.500560    1.289142 6513.073214
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.40 0.43 1.00
act  0.40 1.00 0.36 0.43
ibs  0.43 0.36 1.00 0.43
harv 1.00 0.43 0.43 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-14

	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)	1159.056	12961.392*	10941.553*	1584.936	2632.337	1779.561
	(2245.765)	(998.767)	(2376.976)	(2293.092)	(2870.275)	(2722.928)
SAT	11.945*	.	.	.	14.823	9.384*
	(1.391)				(116.804)	(1.611)
ACT	.	336.313*	.	.	207.63	213.774*
		(43.801)			(128.518)	(49.139)
Iowa BS	.	.	94.757*	.	-23.863	-12.231
			(23.623)		(27.864)	(26.042)
Harvard SS	.	.	.	11.586*	-5.082	.
				(1.402)	(116.868)	
N	485	501	513	459	424	474
RMSE	5012.53	5116.05	5331.74	5045.265	4957.876	4929.418
R^2	0.132	0.106	0.031	0.13	0.158	0.166
adj R^2	0.131	0.104	0.029	0.128	0.15	0.16

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    421 1.0317e+10
2    419 1.0299e+10  2  18250808 0.3712 0.6901
```

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)	1179.038	13074.296*	10958.305*	1324.086	2632.337	1779.561
	(2279.764)	(1023.633)	(2496.314)	(2397.274)	(2870.275)	(2722.928)
SAT	11.943*	.	.	.	14.823	9.384*
	(1.412)				(116.804)	(1.611)
ACT	.	328.936*	.	.	207.63	213.774*
		(44.992)			(128.518)	(49.139)
Iowa BS	.	.	93.881*	.	-23.863	-12.231
			(24.81)		(27.864)	(26.042)
Harvard SS	.	.	.	11.694*	-5.082	.
				(1.465)	(116.868)	
N	474	474	474	424	424	474
RMSE	5018.414	5103.938	5305.316	5017.344	4957.876	4929.418
R^2	0.132	0.102	0.029	0.131	0.158	0.166
adj R^2	0.13	0.1	0.027	0.129	0.15	0.16

$*p \leq 0.05$

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    0.25953505
act    0.19674474
ibs   -0.02165811
```

```
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 0.0602591409
act 0.0335967154
ibs 0.0003915529
```

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%    39.31   40.95   37.60
50%    50.92   50.80   48.48
75%    63.63   61.36   59.77
100%  100.00  100.00  100.00
```

```
mean    51.01    51.22    48.51
sd      18.42    14.63    17.10
var    339.20   214.10   292.50
NA's     0.00     0.00     0.00
N      474.00   474.00   474.00

$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
-16443.8  -3304.0     93.4   3447.2  15424.9

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 13343.215    933.640  14.292  < 2e-16 ***
satpoms        89.638     15.385   5.826 1.05e-08 ***
actpoms        60.541     13.916   4.350 1.67e-05 ***
ibspoms        -8.218     17.498  -0.470    0.639
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4929 on 470 degrees of freedom
Multiple R^2: 0.1657,  Adjusted R^2: 0.1603
F-statistic: 31.11 on 3 and 470 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   0.006199564
act   0.078681149
ibs  -0.041802725
harv -0.002124307
```

```
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  0.000032378428
act  0.005247532110
ibs  0.001474638391
harv 0.000003801484
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-14

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	6330.21*	2914.28
	(2875.629)	(2752.023)
SAT	8.298*	9.289*
	(1.705)	(1.601)
ACT	233.143*	224.525*
	(51.866)	(48.54)
Iowa BS	-16.271	-15.739
	(27.429)	(25.604)
Major: Soc.	.	513.735
		(545.381)
Major: Nat.	.	4411.47*
		(547.696)
Prof. Parents: Yes	.	659.103
		(481.034)
Parent Network: Yes	.	762.943
		(496.607)
Gender: Male	.	-75.414
		(446.516)
N	487	487
RMSE	5267.845	4903.508
R^2	0.137	0.26
adj R^2	0.132	0.248

$*p \leq 0.05$

```
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 =  -103.84009962778 Denominator =  701.165068185468"
```

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

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

```
tval
```

```
difference
-0.1480965
```

```
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
 0.8823291
```

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
     -103.8400996       701.1650682        -0.1480965       478.0000000         0.8823291
```

```
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    483 13403342164
2    478 11493220895  5 1910121269 15.888 1.722e-14 ***
---
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

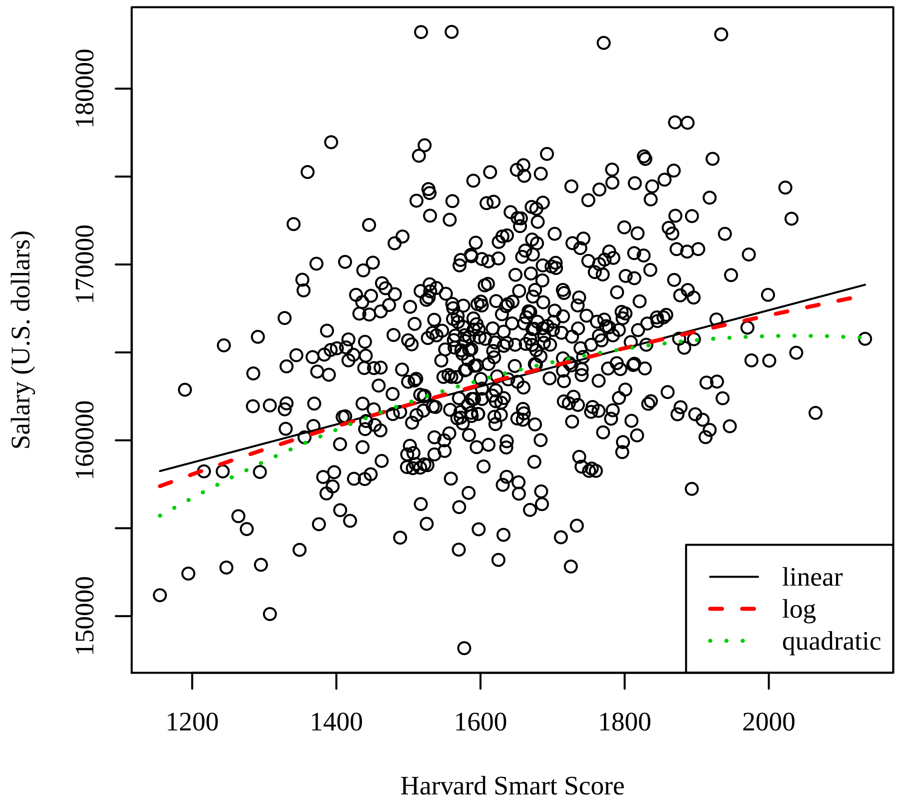
Table 4: Regression with sal3: Student-14

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	145734.303*	32536.145*	111315.109*
	(2324.133)	(16298.529)	(15653.088)
Harvard SS	10.833*	.	53.639*
	(1.375)		(19.302)
Gender: Male	-48.271	-57.696	-96.527
	(456.731)	(455.486)	(455.324)
Major: Soc.	1136.526*	1139.963*	1142.088*
	(552.068)	(550.689)	(549.748)
Major: Nat.	4273.365*	4274.164*	4257.311*
	(560.728)	(559.247)	(558.412)
Prof. Parents: Yes	499.367	514.985	574.492
	(495.355)	(493.882)	(494.424)
Parent Network: Yes	563.39	560.658	545.648
	(512.136)	(510.8)	(510.041)
ln(Harvard SS)	.	17704.728*	.
		(2200.198)	
Harvard SS^2	.	.	-0.013*
			(0.006)
N	472	472	472
RMSE	4940.21	4927.838	4919.396
R^2	0.199	0.203	0.208
adj R^2	0.189	0.193	0.196

$*p \leq 0.05$

```
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%) 21917.48     H
S (30%) 22371.53     S
N (30%) 25686.83     N

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
H (40%) 21917.48      H
S (30%) 22371.53      S
N (30%) 25686.83      N

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

Table 5: Categorical Regressions: Student-14

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21917.477*	22371.528*	2914.28	3428.014
	(392.491)	(412.517)	(2752.023)	(2710.779)
Major: Soc.	454.051	.	513.735	.
	(569.403)		(545.381)	
Major: Nat.	3769.349*	.	4411.47*	.
	(576.646)		(547.696)	
Major 2: Hum.	.	-454.051	.	-513.735
		(569.403)		(545.381)
Major 2: Nat.	.	3315.297*	.	3897.735*
		(590.459)		(554.271)
SAT	.	.	9.289*	9.289*
			(1.601)	(1.601)
ACT	.	.	224.525*	224.525*
			(48.54)	(48.54)
Iowa BS	.	.	-15.739	-15.739
			(25.604)	(25.604)
Prof. Parents: Yes	.	.	659.103	659.103
			(481.034)	(481.034)
Parent Network: Yes	.	.	762.943	762.943
			(496.607)	(496.607)
Gender: Male	.	.	-75.414	-75.414
			(446.516)	(446.516)
N	526	526	487	487
RMSE	5410.116	5410.116	4903.508	4903.508
R^2	0.086	0.086	0.26	0.26
adj R^2	0.083	0.083	0.248	0.248

$*p \leq 0.05$