Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 27
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%	7.54	1045.00	67.14	6091.00	7186.00	146500.00	1033.00
25%	18.88	1517.00	94.01	16870.00	19040.00	161800.00	1499.00
50%	22.38	1610.00	100.20	20670.00	23230.00	165300.00	1595.00
75%	25.76	1731.00	106.60	23800.00	27160.00	169300.00	1714.00
100%	41.67	2095.00	126.90	36170.00	38480.00	181300.00	2060.00
mean	22.30	1618.00	100.20	20360.00	23170.00	165300.00	1600.00
sd	5.05	167.50	9.61	5465.00	5792.00	5642.00	164.10
var	25.49	28050.00	92.35	29860000.00	33550000.00	31830000.00	26940.00
NA's	11.00	56.00	0.00	15.00	0.00	0.00	33.00
N	539.00	539.00	539.00	539.00	539.00	539.00	539.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                   major                    pnet
F             :279.0000   H             :189.0000   NO            :357.0000
M             :260.0000   S             :188.0000   YES           :182.0000
NA's          :  0.0000   N             :162.0000   NA's          :  0.0000
entropy       :  0.9991   NA's          :  0.0000   entropy       :  0.9226
normedEntropy:  0.9991   entropy       :  1.5814   normedEntropy:  0.9226
N             :539.0000   normedEntropy:  0.9978   N             :539.0000
                          N             :539.0000
          pprof
NO            :379.0000
YES           :160.0000
NA's          :  0.0000
entropy       :  0.8774
normedEntropy:  0.8774
N             :539.0000
```

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: 0x2973020>
act ~ sat + ibs + harv
<environment: 0x2973020>
ibs ~ sat + act + harv
<environment: 0x2973020>
harv ~ sat + act + ibs
<environment: 0x2973020>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998638 0.8788224 0.2517995 0.9998673
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
7342.193933    8.252351    1.336540 7533.482789
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.40 0.42 1.00
act  0.40 1.00 0.42 0.43
ibs  0.42 0.42 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-27

	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)	-683.417	11232.79*	7557.637*	241.622	-1836.129	-1827.593
	(2241.761)	(1022.627)	(2435.797)	(2229.895)	(2787.888)	(2691.17)
SAT	13.186*	.	.	.	173.984	9.8*
	(1.394)				(119.708)	(1.575)
ACT	.	408.178*	.	.	437.375*	271.418*
		(44.758)			(133.247)	(50.992)
Iowa BS	.	.	127.769*	.	11.646	4.75
			(24.195)		(27.824)	(26.53)
Harvard SS	.	.	.	12.462*	-164.585	.
				(1.371)	(119.727)	
N	492	513	524	471	434	481
RMSE	5068.199	5088.245	5329.278	4970.832	4842.972	4925.822
R^2	0.154	0.14	0.051	0.15	0.215	0.21
adj R^2	0.153	0.138	0.049	0.148	0.208	0.205

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    431 1.0110e+10
2    429 1.0062e+10  2  47571159 1.0141 0.3636
```

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)	-934.184	11282.302*	7112.096*	-590.681	-1836.129	-1827.593
	(2259.608)	(1065.084)	(2530.2)	(2308.851)	(2787.888)	(2691.17)
SAT	13.336*	.	.	.	173.984	9.8*
	(1.407)				(119.708)	(1.575)
ACT	.	406.67*	.	.	437.375*	271.418*
		(46.482)			(133.247)	(50.992)
Iowa BS	.	.	132.341*	.	11.646	4.75
			(25.133)		(27.824)	(26.53)
Harvard SS	.	.	.	12.976*	-164.585	.
				(1.42)	(119.727)	
N	481	481	481	434	434	481
RMSE	5074.841	5135.287	5376.958	4986.943	4842.972	4925.822
R^2	0.158	0.138	0.055	0.162	0.215	0.21
adj R^2	0.156	0.136	0.053	0.16	0.208	0.205

$*p \leq 0.05$

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   0.273920610
act   0.236782951
ibs   0.008197129
```

```
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 6.408415e-02
act 4.692313e-02
ibs 5.308606e-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%    33.31   44.98   45.26
50%    43.69   55.34   54.53
75%    53.65   65.87   65.70
100%  100.00  100.00  100.00
```

```
mean    43.40    55.36    54.97
sd      14.77    16.35    16.03
var    218.30   267.40   256.90
NA's     0.00     0.00     0.00
N      481.00   481.00   481.00

$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
-14037  -3002     111    3224   15768

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 10661.190    968.150  11.012  < 2e-16 ***
satpoms       100.665     16.183   6.220 1.09e-09 ***
actpoms        92.635     17.403   5.323 1.58e-07 ***
ibspoms         2.837     15.844   0.179    0.858
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4926 on 477 degrees of freedom
Multiple R^2:  0.21,    Adjusted R^2: 0.205
F-statistic: 42.27 on 3 and 477 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   0.06999863
act   0.15652392
ibs   0.02020472
harv -0.06622400
```

```
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  0.0038648347
act  0.0197129987
ibs  0.0003205543
harv 0.0034574690
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-27

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	96.775	-2338.05
	(2861.43)	(2704.607)
SAT	9.129*	9.491*
	(1.659)	(1.554)
ACT	274.481*	289.562*
	(53.661)	(50.413)
Iowa BS	23.612	10.101
	(28.055)	(26.242)
Major: Soc.	.	2042.9*
		(535.544)
Major: Nat.	.	4595.727*
		(550.786)
Prof. Parents: Yes	.	687.701
		(487.212)
Parent Network: Yes	.	603.446
		(472.742)
Gender: Male	.	702.827
		(445.076)
N	495	495
RMSE	5290.698	4937.402
R^2	0.189	0.301
adj R^2	0.185	0.29

$*p \leq 0.05$

```
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 =  84.254961690606 Denominator =  695.620383317092"
```

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

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

```
tval
```

```
difference
  0.121122
```

```
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
 0.9036444
```

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
      84.2549617       695.6203833       0.1211220      486.0000000        0.9036444
```

```
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    491 13743819848
2    486 11847677722  5 1896142126 15.556 3.289e-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

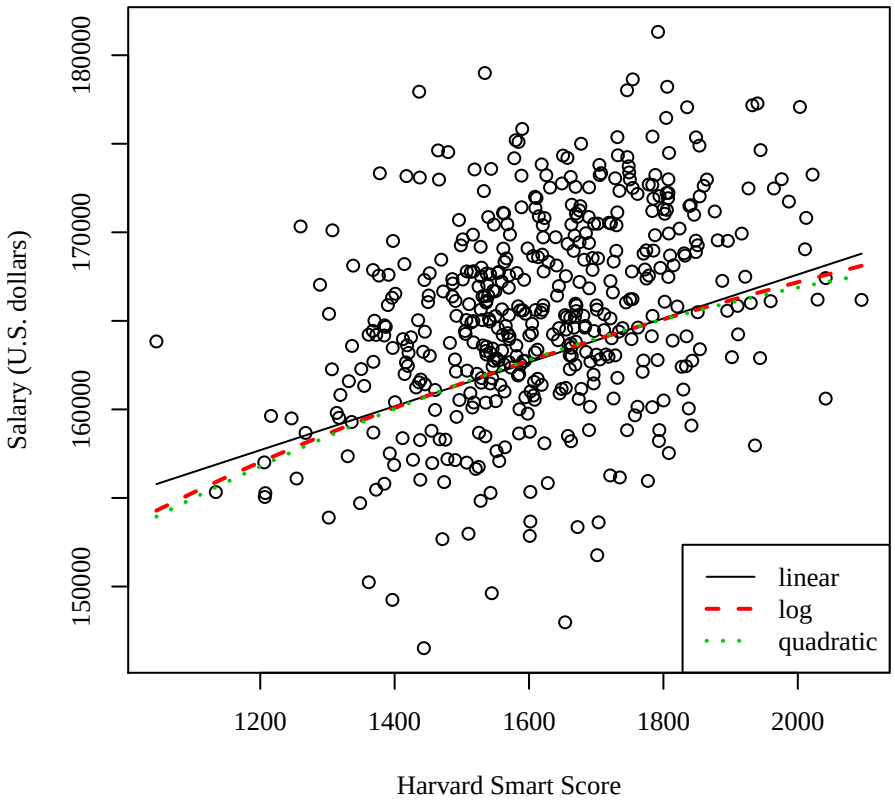
Table 4: Regression with sal3: Student-27

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	142823.46*	16077.225	127178.668*
	(2185.188)	(15456.949)	(14471.861)
Harvard SS	12.397*	.	31.906
	(1.312)		(17.888)
Gender: Male	288.128	292.16	293.879
	(438.663)	(438.252)	(438.604)
Major: Soc.	2673.918*	2719.959*	2736.971*
	(525.563)	(525.314)	(528.609)
Major: Nat.	4977.429*	4982.498*	4984.159*
	(542.81)	(542.296)	(542.733)
Prof. Parents: Yes	1252.354*	1253.553*	1256.243*
	(478.844)	(478.37)	(478.759)
Parent Network: Yes	-1017.761*	-1021.997*	-1022.145*
	(465.53)	(465.073)	(465.451)
ln(Harvard SS)	.	19880.19*	.
		(2090.701)	
Harvard SS^2	.	.	-0.006
			(0.006)
N	483	483	483
RMSE	4804.499	4799.901	4803.51
R^2	0.287	0.288	0.289
adj R^2	0.278	0.279	0.278

$*p \leq 0.05$

```
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%) 21201.27     H
S (30%) 23069.77     S
N (30%) 25567.42     N

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
H (40%) 21201.27      H
S (30%) 23069.77      S
N (30%) 25567.42      N

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

Table 5: Categorical Regressions: Student-27

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21201.267* (402.167)	23069.771* (403.235)	-2338.05 (2704.607)	-295.151 (2705.709)
Major: Soc.	1868.504* (569.505)	.	2042.9* (535.544)	.
Major: Nat.	4366.15* (591.973)	.	4595.727* (550.786)	.
Major 2: Hum.	.	-1868.504* (569.505)	.	-2042.9* (535.544)
Major 2: Nat.	.	2497.647* (592.7)	.	2552.827* (551.759)
SAT	.	.	9.491* (1.554)	9.491* (1.554)
ACT	.	.	289.562* (50.413)	289.562* (50.413)
Iowa BS	.	.	10.101 (26.242)	10.101 (26.242)
Prof. Parents: Yes	.	.	687.701 (487.212)	687.701 (487.212)
Parent Network: Yes	.	.	603.446 (472.742)	603.446 (472.742)
Gender: Male	.	.	702.827 (445.076)	702.827 (445.076)
N	539	539	495	495
RMSE	5528.878	5528.878	4937.402	4937.402
R^2	0.092	0.092	0.301	0.301
adj R^2	0.089	0.089	0.29	0.29

$*p \leq 0.05$