Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 21
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%	6.49	1135.00	72.03	882.50	984.80	147400.00	1101.00
25%	18.76	1516.00	94.16	17240.00	19500.00	161000.00	1496.00
50%	22.19	1631.00	100.30	20270.00	23050.00	164900.00	1605.00
75%	25.26	1731.00	107.70	23740.00	27320.00	168900.00	1709.00
100%	38.17	2256.00	130.80	34320.00	38810.00	185300.00	2232.00
mean	22.14	1625.00	100.70	20350.00	23210.00	165100.00	1601.00
sd	5.08	163.50	9.95	5151.00	5680.00	5912.00	162.90
var	25.76	26740.00	99.03	26530000.00	32260000.00	34950000.00	26540.00
NA's	14.00	50.00	0.00	9.00	0.00	0.00	19.00
N	558.00	558.00	558.00	558.00	558.00	558.00	558.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
            gender                     major                     pnet
M            :284.0000   S            :196.0000   NO           :391.0000
F            :274.0000   H            :192.0000   YES          :167.0000
NA's         :  0.0000   N            :170.0000   NA's         :  0.0000
entropy      :  0.9998   NA's         :  0.0000   entropy      :  0.8804
normedEntropy:  0.9998   entropy      :  1.5822   normedEntropy:  0.8804
N            :558.0000   normedEntropy:  0.9983   N            :558.0000
                         N            :558.0000
            pprof
NO           :383.0000
YES          :175.0000
NA's         :  0.0000
entropy      :  0.8973
normedEntropy:  0.8973
N            :558.0000
```

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: 0x13be9e8>
act ~ sat + ibs + harv
<environment: 0x13be9e8>
ibs ~ sat + act + harv
<environment: 0x13be9e8>
harv ~ sat + act + ibs
<environment: 0x13be9e8>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998267 0.8582965 0.2141362 0.9998308
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
5770.573486    7.056987    1.272485 5910.893589
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.38 0.39  1.0
act  0.38 1.00 0.37  0.4
ibs  0.39 0.37 1.00  0.4
harv 1.00 0.40 0.40  1.0
```

```
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-21

	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)	5049.028*	14713.099*	14127.412*	3251.246	6036.37*	6859.142*
	(2134.054)	(955.77)	(2227.78)	(2190.737)	(2724.328)	(2606.334)
SAT	9.52*	.	.	.	52.965	7.539*
	(1.326)				(106.389)	(1.498)
ACT	.	254.991*	.	.	203.772	177.339*
		(42.099)			(116.757)	(47.069)
Iowa BS	.	.	61.794*	.	-33.114	-25.404
			(22.014)		(25.32)	(24.422)
Harvard SS	.	.	.	10.547*	-44.179	.
				(1.342)	(106.38)	
N	531	535	549	500	471	517
RMSE	4929.368	4961.037	5118.645	4867.045	4852.523	4870.194
R^2	0.089	0.064	0.014	0.11	0.116	0.105
adj R^2	0.087	0.063	0.012	0.109	0.108	0.099

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    468 1.1016e+10
2    466 1.0973e+10  2  43475287 0.9232  0.398
```

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)	5886.426*	14826.591*	14616.288*	4271.817	6036.37*	6859.142*
	(2166.326)	(977.855)	(2298.943)	(2272.924)	(2724.328)	(2606.334)
SAT	8.997*	.	.	.	52.965	7.539*
	(1.346)				(106.389)	(1.498)
ACT	.	247.303*	.	.	203.772	177.339*
		(43.131)			(116.757)	(47.069)
Iowa BS	.	.	56.378*	.	-33.114	-25.404
			(22.731)		(25.32)	(24.422)
Harvard SS	.	.	.	9.887*	-44.179	.
				(1.394)	(106.38)	
N	517	517	517	471	471	517
RMSE	4927.529	4980.238	5106.334	4887.587	4852.523	4870.194
R^2	0.08	0.06	0.012	0.097	0.116	0.105
adj R^2	0.078	0.058	0.01	0.095	0.108	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  -1.0000000
sat    0.2168976
act    0.1640909
ibs   -0.0458767
```

```
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  0.044204306
act  0.024777103
ibs  0.001888543
```

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%    38.45    37.84    35.16
50%    49.37    48.01    44.65
75%    59.22    60.67    53.64
100%  100.00   100.00   100.00
```

```
mean    49.26    48.70    44.23
sd      16.05    16.82    14.24
var    257.50   283.10   202.90
NA's     0.00     0.00     0.00
N      517.00   517.00   517.00

$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
-14906.9  -2976.5     75.1   3198.1  12892.2

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 14478.31     875.66  16.534  < 2e-16 ***
satpoms        85.28      16.95   5.032 6.71e-07 ***
actpoms        56.18      14.91   3.768 0.000184 ***
ibspoms       -14.93      14.36  -1.040 0.298746
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4870 on 513 degrees of freedom
Multiple R^2: 0.1046,  Adjusted R^2: 0.09934
F-statistic: 19.97 on 3 and 513 DF,  p-value: 2.948e-12
```

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   0.02305587
act   0.08058468
ibs  -0.06047343
harv -0.01923474
```

```
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  0.0004704159
act  0.0057812590
ibs  0.0032464476
harv 0.0003273570
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-21

	Test Scores Only	All Predictors
	Estimate	Estimate
	(S.E.)	(S.E.)
(Intercept)	10280.985*	6499.997*
	(2842.651)	(2612.4)
SAT	6.908*	7.477*
	(1.632)	(1.475)
ACT	215.604*	179.759*
	(51.743)	(46.944)
Iowa BS	-29.315	-26.1
	(26.84)	(24.411)
Major: Soc.	.	2286.65*
		(515.443)
Major: Nat.	.	5666.697*
		(532.607)
Prof. Parents: Yes	.	1089.937*
		(463.193)
Parent Network: Yes	.	377.745
		(470.782)
Gender: Male	.	708.324
		(427.419)
N	525	525
RMSE	5402.891	4875.283
R^2	0.092	0.267
adj R^2	0.086	0.256
$*p \leq 0.05$		

```
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 =  712.191787190729 Denominator =  677.93721187849"
```

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

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

```
tval
```

```
difference
  1.050528
```

```
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
 0.2939673
```

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
      712.1917872       677.9372119         1.0505277       516.0000000         0.2939673
```

```
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    521 15208632419
2    516 12264485860  5 2944146559 24.774 < 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

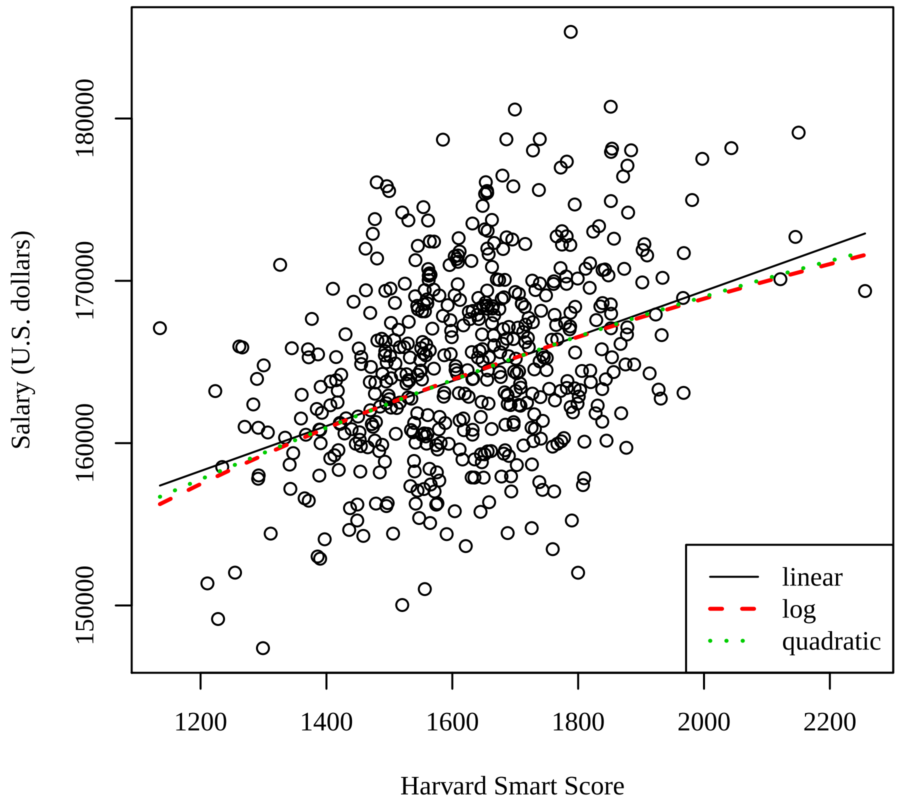
Table 4: Regression with sal3: Student-21

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	140516.345*	-2147.609	132501.015*
	(2292.578)	(16364.796)	(14777.868)
Harvard SS	13.859*	.	23.769
	(1.374)		(18.102)
Gender: Male	-642.088	-639.019	-639.593
	(449.629)	(449.49)	(449.966)
Major: Soc.	1776.426*	1748.846*	1756.035*
	(536.148)	(535.934)	(537.806)
Major: Nat.	4073.886*	4076.699*	4076.942*
	(562.805)	(562.651)	(563.225)
Prof. Parents: Yes	2132.531*	2152.685*	2149.081*
	(487.433)	(487.233)	(488.704)
Parent Network: Yes	-585.439	-571.139	-576.609
	(491.7)	(491.581)	(492.306)
ln(Harvard SS)	.	22357.624*	.
		(2213.071)	
Harvard SS^2	.	.	-0.003
			(0.006)
N	508	508	508
RMSE	5049.888	5048.393	5053.412
R^2	0.262	0.263	0.263
adj R^2	0.254	0.254	0.253

$*p \leq 0.05$

```
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%) 22971.90     S
H (30%) 20612.49     H
N (30%) 26418.55     N

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

```
predictOMatic(cm2)
```

```
$major2
              fit major2
S (40%) 22971.90      S
H (30%) 20612.49      H
N (30%) 26418.55      N

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

Table 5: Categorical Regressions: Student-21

	major	major2	major full	major2 full
	Estimate	Estimate	Estimate	Estimate
	(S.E.)	(S.E.)	(S.E.)	(S.E.)
(Intercept)	20612.488*	22971.902*	6499.997*	8786.647*
	(374.074)	(370.237)	(2612.4)	(2626.08)
Major: Soc.	2359.414*	.	2286.65*	.
	(526.315)		(515.443)	
Major: Nat.	5806.061*	.	5666.697*	.
	(545.868)		(532.607)	
Major 2: Hum.	.	-2359.414*	.	-2286.65*
		(526.315)		(515.443)
Major 2: Nat.	.	3446.647*	.	3380.048*
		(543.246)		(528.41)
SAT	.	.	7.477*	7.477*
			(1.475)	(1.475)
ACT	.	.	179.759*	179.759*
			(46.944)	(46.944)
Iowa BS	.	.	-26.1	-26.1
			(24.411)	(24.411)
Prof. Parents: Yes	.	.	1089.937*	1089.937*
			(463.193)	(463.193)
Parent Network: Yes	.	.	377.745	377.745
			(470.782)	(470.782)
Gender: Male	.	.	708.324	708.324
			(427.419)	(427.419)
N	558	558	525	525
RMSE	5183.325	5183.325	4875.283	4875.283
R^2	0.17	0.17	0.267	0.267
adj R^2	0.167	0.167	0.256	0.256

$*p \leq 0.05$