Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 10
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%	3.47	1085.00	76.32	5782.00	6815.00	147800.00	1069.00
25%	18.85	1511.00	93.87	16590.00	19210.00	161900.00	1485.00
50%	22.28	1622.00	99.95	20380.00	23410.00	165600.00	1598.00
75%	25.83	1737.00	106.00	23930.00	27280.00	169600.00	1719.00
100%	36.85	2096.00	128.80	35260.00	38220.00	182400.00	2070.00
mean	22.25	1623.00	100.00	20370.00	23300.00	165700.00	1602.00
sd	5.09	157.60	9.83	5020.00	5514.00	5805.00	159.30
var	25.94	24830.00	96.57	25200000.00	30410000.00	33700000.00	25380.00
NA's	17.00	43.00	0.00	15.00	0.00	0.00	30.00
N	542.00	542.00	542.00	542.00	542.00	542.00	542.00

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            :205.0000   NO           :381.0000
F            :258.0000   N            :173.0000   YES          :161.0000
NA's         :  0.0000   H            :164.0000   NA's         :  0.0000
entropy      :  0.9983   NA's         :  0.0000   entropy      :  0.8777
normedEntropy:  0.9983   entropy      :  1.5782   normedEntropy:  0.8777
N            :542.0000   normedEntropy:  0.9958   N            :542.0000
                         N            :542.0000
          pprof
NO           :381.0000
YES          :161.0000
NA's         :  0.0000
entropy      :  0.8777
normedEntropy:  0.8777
N            :542.0000
```

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: 0x267d0a0>
act ~ sat + ibs + harv
<environment: 0x267d0a0>
ibs ~ sat + act + harv
<environment: 0x267d0a0>
harv ~ sat + act + ibs
<environment: 0x267d0a0>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998262 0.8546890 0.2338246 0.9998298
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
5752.669914    6.881790    1.305184 5875.722323
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.31 0.41 1.00
act  0.31 1.00 0.36 0.34
ibs  0.41 0.36 1.00 0.42
harv 1.00 0.34 0.42 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-10

	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)	2742.672	15464.656*	13307.99*	824.101	1620.597	3418.985
	(2177.81)	(971.054)	(2222.955)	(2187.836)	(2673.025)	(2587.694)
SAT	11.034*	.	.	.	-1.02	10.121*
	(1.353)				(105.855)	(1.519)
ACT	.	219.975*	.	.	122.551	146.18*
		(42.466)			(112.13)	(45.592)
Iowa BS	.	.	70.503*	.	-29.227	-24.825
			(22.101)		(25.363)	(24.769)
Harvard SS	.	.	.	12.005*	12.647	.
				(1.342)	(105.752)	
N	497	510	527	486	444	482
RMSE	4771.437	4895.537	4977.098	4618.521	4608.715	4719.371
R^2	0.118	0.05	0.019	0.142	0.163	0.138
adj R^2	0.117	0.048	0.017	0.14	0.156	0.132

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    441 9352705886
2    439 9324469673  2  28236213 0.6647  0.515
```

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)	2805.429	15314.818*	13351.385*	501.091	1620.597	3418.985
	(2196.723)	(997.943)	(2321.243)	(2261.913)	(2673.025)	(2587.694)
SAT	10.992*	.	.	.	-1.02	10.121*
	(1.364)				(105.855)	(1.519)
ACT	.	228.514*	.	.	122.551	146.18*
		(43.516)			(112.13)	(45.592)
Iowa BS	.	.	70.648*	.	-29.227	-24.825
			(23.084)		(25.363)	(24.769)
Harvard SS	.	.	.	12.217*	12.647	.
				(1.386)	(105.752)	
N	482	482	482	444	444	482
RMSE	4760.018	4932.087	5023.013	4630.335	4608.715	4719.371
R^2	0.119	0.054	0.019	0.149	0.163	0.138
adj R^2	0.117	0.052	0.017	0.148	0.156	0.132

$*p \leq 0.05$

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    0.29153248
act    0.14509922
ibs   -0.04579393
```

```
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  0.080090734
act  0.018544067
ibs  0.001812012
```

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%     46.12    33.48    41.79
50%     56.58    45.89    52.83
75%     67.49    57.11    64.79
100%   100.00   100.00   100.00
```

```
mean    56.54    45.28    53.28
sd      15.48    18.92    15.89
var    239.70   358.00   252.50
NA's     0.00     0.00     0.00
N      482.00   482.00   482.00

$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
-12059.0   -3059.3    -324.7    2949.9   14102.9

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 12851.85     957.52  13.422  < 2e-16 ***
satpoms       101.35      15.21   6.663 7.39e-11 ***
actpoms        48.79      15.22   3.206  0.00143 **
ibspoms       -13.02      12.99  -1.002  0.31673
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4719 on 478 degrees of freedom
Multiple R^2: 0.1377,   Adjusted R^2: 0.1323
F-statistic: 25.45 on 3 and 478 DF,  p-value: 2.716e-15
```

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  -0.0004596774
act   0.0520919712
ibs  -0.0549151638
harv  0.0057075660
```

```
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  0.0000001768328
act  0.0022770796939
ibs  0.0025313534577
harv 0.0000272629340
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-10

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	6923.862*	3910.26
	(2832.486)	(2604.156)
SAT	9.4*	10.456*
	(1.667)	(1.507)
ACT	161.966*	153.489*
	(50.537)	(45.906)
Iowa BS	-22.433	-36.267
	(27.341)	(24.824)
Major: Soc.	.	1541.235*
		(523.505)
Major: Nat.	.	5330.899*
		(543.888)
Prof. Parents: Yes	.	644.693
		(467.718)
Parent Network: Yes	.	1745.89*
		(470.044)
Gender: Male	.	-186.818
		(429.627)
N	497	497
RMSE	5273.423	4753.878
R^2	0.11	0.284
adj R^2	0.104	0.272

$*p \leq 0.05$

```
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 =  -1101.19722444185 Denominator =  670.009001796839"
```

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

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

```
tval
```

```
difference
 -1.643556
```

```
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
 0.1009121
```

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
    -1101.1972244       670.0090018        -1.6435559       488.0000000         0.1009121
```

```
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    493 13709834292
2    488 11028487689  5 2681346602 23.729 < 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

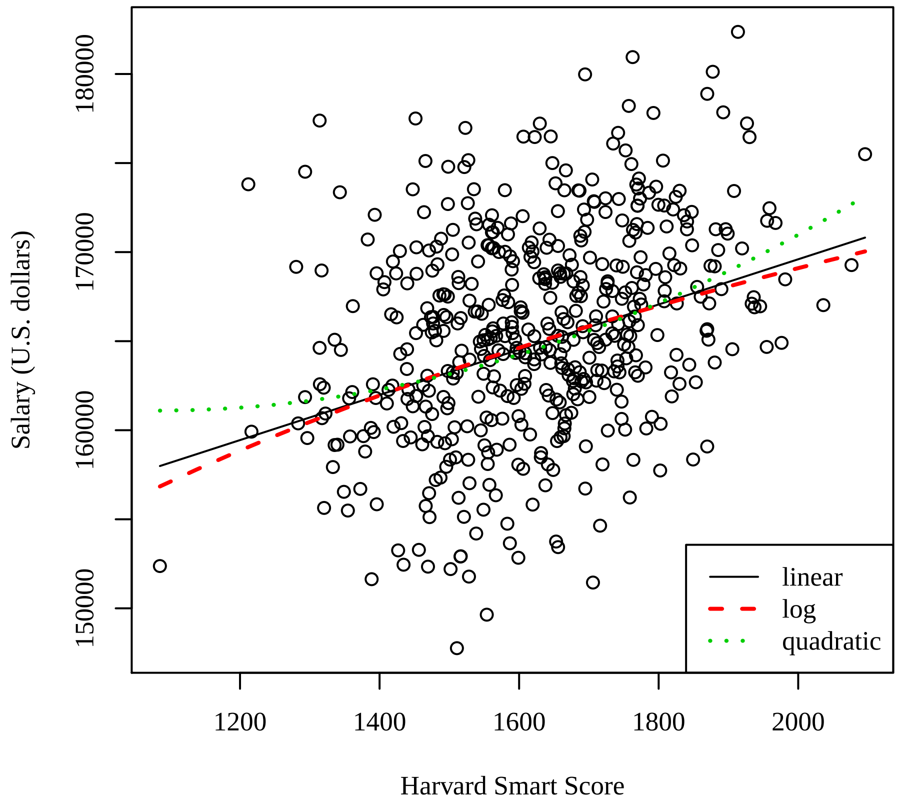
Table 4: Regression with sal3: Student-10

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	142547.705*	15047.195	173154.926*
	(2451.698)	(17502.669)	(18014.275)
Harvard SS	12.691*	.	-25.404
	(1.472)		(22.262)
Gender: Male	-44.977	-42.707	-55.307
	(467.136)	(468.224)	(466.256)
Major: Soc.	1714.089*	1706.474*	1751.608*
	(565.097)	(566.443)	(564.41)
Major: Nat.	4658.443*	4649.441*	4681.443*
	(585.985)	(587.325)	(584.986)
Prof. Parents: Yes	1286.852*	1253.846*	1384.038*
	(511.888)	(513.042)	(514.014)
Parent Network: Yes	135.093	152.382	76.273
	(507.208)	(508.391)	(507.371)
ln(Harvard SS)	.	20048.921*	.
		(2367.931)	
Harvard SS^2	.	.	0.012
			(0.007)
N	499	499	499
RMSE	5168.112	5180.129	5157.948
R^2	0.222	0.218	0.226
adj R^2	0.212	0.209	0.215

$*p \leq 0.05$

```
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%) 22849.31      S
N (30%) 26070.49      N
H (30%) 20948.80      H

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

```
predictOMatic(cm2)
```

```
$major2
              fit  major2
S (40%) 22849.31       S
N (30%) 26070.49       N
H (30%) 20948.80       H

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

Table 5: Categorical Regressions: Student-10

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	20948.802*	22849.306*	3910.26	5451.494*
	(400.422)	(358.149)	(2604.156)	(2618.985)
Major: Soc.	1900.504*	.	1541.235*	.
	(537.223)		(523.505)	
Major: Nat.	5121.684*	.	5330.899*	.
	(558.869)		(543.888)	
Major 2: Hum.	.	-1900.504*	.	-1541.235*
		(537.223)		(523.505)
Major 2: Nat.	.	3221.18*	.	3789.664*
		(529.403)		(515.437)
SAT	.	.	10.456*	10.456*
			(1.507)	(1.507)
ACT	.	.	153.489*	153.489*
			(45.906)	(45.906)
Iowa BS	.	.	-36.267	-36.267
			(24.824)	(24.824)
Prof. Parents: Yes	.	.	644.693	644.693
			(467.718)	(467.718)
Parent Network: Yes	.	.	1745.89*	1745.89*
			(470.044)	(470.044)
Gender: Male	.	.	-186.818	-186.818
			(429.627)	(429.627)
N	542	542	497	497
RMSE	5127.908	5127.908	4753.878	4753.878
R^2	0.138	0.138	0.284	0.284
adj R^2	0.135	0.135	0.272	0.272

$*p \leq 0.05$