Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 41
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%	5.04	1171.00	66.03	4467.00	7629.00	148100.00	1161.00
25%	18.44	1492.00	93.25	17040.00	19100.00	161300.00	1468.00
50%	21.65	1615.00	99.99	20720.00	23380.00	165400.00	1590.00
75%	25.78	1717.00	106.70	24050.00	27230.00	168900.00	1696.00
100%	35.98	2019.00	136.20	36450.00	40190.00	186700.00	2059.00
mean	22.00	1607.00	99.88	20410.00	23200.00	165200.00	1585.00
sd	4.97	166.30	10.44	5281.00	5747.00	5897.00	164.00
var	24.71	27660.00	108.90	27890000.00	33030000.00	34780000.00	26900.00
NA's	24.00	52.00	0.00	16.00	0.00	0.00	19.00
N	561.00	561.00	561.00	561.00	561.00	561.00	561.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
           gender                       major                      pnet
F            :289.0000   H            :204.0000   NO           :376.0000
M            :272.0000   N            :180.0000   YES          :185.0000
NA's         :  0.0000   S            :177.0000   NA's         :  0.0000
entropy      :  0.9993   NA's         :  0.0000   entropy      :  0.9147
normedEntropy:  0.9993   entropy      :  1.5820   normedEntropy:  0.9147
N            :561.0000   normedEntropy:  0.9981   N            :561.0000
                         N            :561.0000
           pprof
NO           :386.0000
YES          :175.0000
NA's         :  0.0000
entropy      :  0.8954
normedEntropy:  0.8954
N            :561.0000
```

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: 0x20757a0>
act ~ sat + ibs + harv
<environment: 0x20757a0>
ibs ~ sat + act + harv
<environment: 0x20757a0>
harv ~ sat + act + ibs
<environment: 0x20757a0>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998467 0.8605251 0.2493557 0.9998509
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
6522.474361    7.169751    1.332189 6708.307744
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.46 0.46 1.00
act  0.46 1.00 0.38 0.49
ibs  0.46 0.38 1.00 0.47
harv 1.00 0.49 0.47 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-41

	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)	-358.231	13292.846*	9446.95*	-774.276	123.857	-722.388
	(2045.173)	(1006.369)	(2112.534)	(2117.506)	(2522.389)	(2430.33)
SAT	13.048*	.	.	.	3.793	10.911*
	(1.282)				(108.461)	(1.563)
ACT	.	324.998*	.	.	181.748	156.124*
		(44.619)			(120.361)	(49.455)
Iowa BS	.	.	109.755*	.	-10.618	3.328
			(21.033)		(24.565)	(23.647)
Harvard SS	.	.	.	13.147*	6.997	.
				(1.309)	(108.546)	
N	526	522	545	495	457	504
RMSE	4799.004	5071.985	5157.969	4829.81	4764.659	4783.257
R^2	0.165	0.093	0.048	0.17	0.193	0.185
adj R^2	0.163	0.091	0.046	0.168	0.186	0.18

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    454 1.0266e+10
2    452 1.0261e+10  2   4344032 0.0957 0.9088
```

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)	-566.217	13255.609*	9231.065*	-653.695	123.857	-722.388
	(2095.02)	(1017.143)	(2219.316)	(2164.229)	(2522.389)	(2430.33)
SAT	13.181*	.	.	.	3.793	10.911*
	(1.312)				(108.461)	(1.563)
ACT	.	323.903*	.	.	181.748	156.124*
		(45.151)			(120.361)	(49.455)
Iowa BS	.	.	111.358*	.	-10.618	3.328
			(22.062)		(24.565)	(23.647)
Harvard SS	.	.	.	13.037*	6.997	.
				(1.338)	(108.546)	
N	504	504	504	457	457	504
RMSE	4824.451	5035.538	5158.089	4809.28	4764.659	4783.257
R^2	0.167	0.093	0.048	0.173	0.193	0.185
adj R^2	0.166	0.091	0.046	0.171	0.186	0.18

$*p \leq 0.05$

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    0.298062744
act    0.139795657
ibs    0.006292792
```

```
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 0.0794799533
act 0.0162478132
ibs 0.0000322805
```

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%    43.28   33.12   34.93
50%    53.68   43.60   48.00
75%    67.07   54.17   59.69
100%  100.00  100.00  100.00
```

```
mean    54.73   43.62   47.60
sd      16.07   16.24   18.27
var    258.30  263.80  333.90
NA's     0.00    0.00    0.00
N      504.00  504.00  504.00

$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
-14903.0  -2990.9    305.8   3533.5  13028.6

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 12976.016    834.371  15.552  < 2e-16 ***
satpoms        97.905     14.022   6.982 9.29e-12 ***
actpoms        48.305     15.301   3.157  0.00169 **
ibspoms         2.136     15.177   0.141  0.88815
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4783 on 500 degrees of freedom
Multiple R^2: 0.1849,  Adjusted R^2:  0.18
F-statistic:  37.8 on 3 and 500 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   0.001644842
act   0.070846782
ibs  -0.020326314
harv  0.003031992
```

```
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  0.000002182435
act  0.004069279058
ibs  0.000333417682
harv 0.000007415695
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-41

	Test Scores Only	All Predictors
	Estimate	Estimate
	(S.E.)	(S.E.)
(Intercept)	1511.63	-408.046
	(2632.135)	(2427.007)
SAT	11.404*	10.994*
	(1.669)	(1.53)
ACT	176.084*	146.564*
	(53.482)	(49.216)
Iowa BS	-3.317	-0.495
	(25.656)	(23.507)
Major: Soc.	.	1502.429*
		(509.518)
Major: Nat.	.	4365.906*
		(515.431)
Prof. Parents: Yes	.	1573.881*
		(458.079)
Parent Network: Yes	.	1649.273*
		(448.499)
Gender: Male	.	90.709
		(422.187)
N	519	519
RMSE	5235.448	4780.566
R^2	0.173	0.317
adj R^2	0.168	0.306

$*p \leq 0.05$

```
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 =  -75.3918717361005 Denominator =  627.331249006599"
```

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

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

```
tval
```

```
difference
-0.1201787
```

```
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
 0.9043889
```

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
      -75.3918717       627.3312490        -0.1201787       510.0000000         0.9043889
```

```
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    515 14116108606
2    510 11655443347  5 2460665260 21.534 < 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

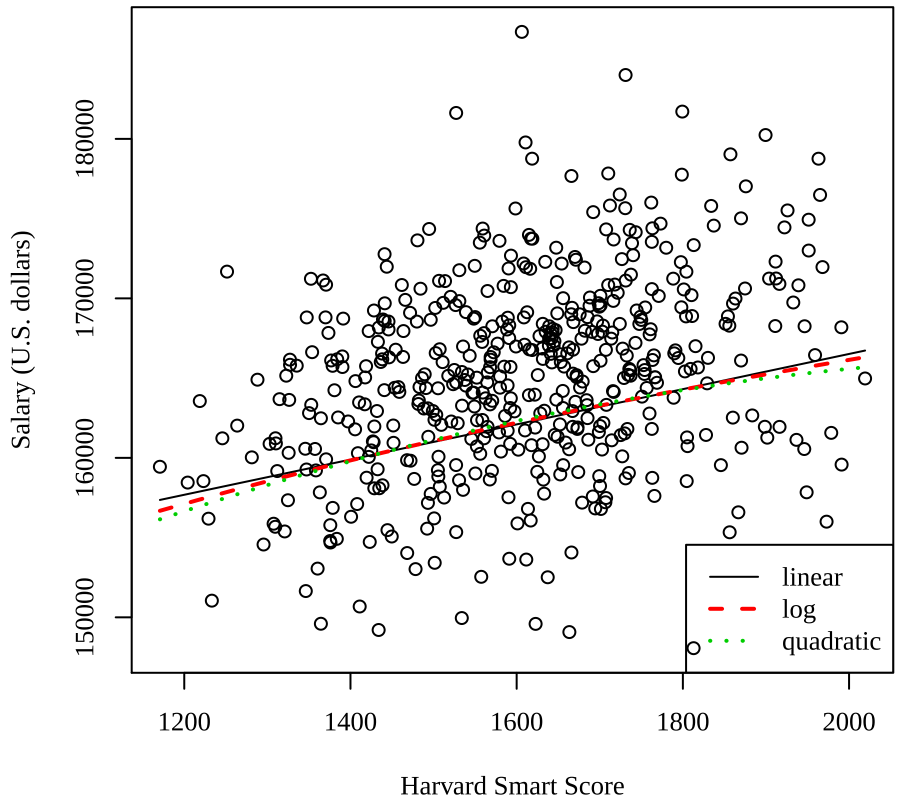
Table 4: Regression with sal3: Student-41

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	144438.641*	31832.828*	125283.94*
	(2253.653)	(16166.321)	(16956.913)
Harvard SS	11.038*	.	35.129
	(1.381)		(21.183)
Gender: Male	-67.546	-53.737	-38.998
	(460.937)	(460.582)	(461.48)
Major: Soc.	2777.319*	2763.22*	2743.249*
	(556.671)	(556.209)	(557.308)
Major: Nat.	5726.994*	5706.564*	5683.214*
	(562.886)	(562.475)	(564.028)
Prof. Parents: Yes	1003.641*	997.155*	991.783*
	(497.516)	(497.104)	(497.476)
Parent Network: Yes	-31.261	-27.597	-23.042
	(489.66)	(489.204)	(489.567)
ln(Harvard SS)	.	17670.318*	.
		(2192.743)	
Harvard SS^2	.	.	-0.007
			(0.007)
N	509	509	509
RMSE	5159.029	5154.422	5157.493
R^2	0.268	0.27	0.27
adj R^2	0.26	0.261	0.26

$*p \leq 0.05$

```
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%) 21227.55     H
N (30%) 26113.72     N
S (30%) 22524.78     S

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

```
predictOMatic(cm2)
```

```
$major2
              fit major2
H (40%) 21227.55      H
N (30%) 26113.72      N
S (30%) 22524.78      S

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

Table 5: Categorical Regressions: Student-41

	major	major2	major full	major2 full
	Estimate	Estimate	Estimate	Estimate
	(S.E.)	(S.E.)	(S.E.)	(S.E.)
(Intercept)	21227.55*	22524.782*	-408.046	1094.383
	(376.016)	(403.678)	(2427.007)	(2441.264)
Major: Soc.	1297.232*	.	1502.429*	.
	(551.674)		(509.518)	
Major: Nat.	4886.167*	.	4365.906*	.
	(549.207)		(515.431)	
Major 2: Hum.	.	-1297.232*	.	-1502.429*
		(551.674)		(509.518)
Major 2: Nat.	.	3588.935*	.	2863.478*
		(568.503)		(532.174)
SAT	.	.	10.994*	10.994*
			(1.53)	(1.53)
ACT	.	.	146.564*	146.564*
			(49.216)	(49.216)
Iowa BS	.	.	-0.495	-0.495
			(23.507)	(23.507)
Prof. Parents: Yes	.	.	1573.881*	1573.881*
			(458.079)	(458.079)
Parent Network: Yes	.	.	1649.273*	1649.273*
			(448.499)	(448.499)
Gender: Male	.	.	90.709	90.709
			(422.187)	(422.187)
N	561	561	519	519
RMSE	5370.584	5370.584	4780.566	4780.566
R^2	0.13	0.13	0.317	0.317
adj R^2	0.127	0.127	0.306	0.306

$*p \leq 0.05$