Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 30
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%	6.04	1197.00	59.78	3836.00	6014.00	148300.00	1058.00
25%	18.85	1524.00	94.62	17000.00	19280.00	162000.00	1503.00
50%	22.40	1629.00	101.50	20500.00	23150.00	165400.00	1606.00
75%	26.05	1748.00	107.50	24100.00	27170.00	169100.00	1721.00
100%	38.18	2086.00	129.70	36320.00	41450.00	181200.00	2057.00
mean	22.46	1634.00	101.20	20510.00	23310.00	165500.00	1608.00
sd	5.31	157.30	10.34	5564.00	5958.00	5378.00	159.40
var	28.23	24730.00	106.90	30960000.00	35500000.00	28920000.00	25410.00
NA's	17.00	45.00	0.00	12.00	0.00	0.00	36.00
N	541.00	541.00	541.00	541.00	541.00	541.00	541.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                   major                    pnet
M             :276.0000   H             :201.0000   NO            :369.0000
F             :265.0000   S             :175.0000   YES           :172.0000
NA's          :  0.0000   N             :165.0000   NA's          :  0.0000
entropy       :  0.9997   NA's          :  0.0000   entropy       :  0.9021
normedEntropy:  0.9997   entropy       :  1.5799   normedEntropy:  0.9021
N             :541.0000   normedEntropy:  0.9968   N             :541.0000
                          N             :541.0000
          pprof
NO            :373.0000
YES           :168.0000
NA's          :  0.0000
entropy       :  0.8938
normedEntropy:  0.8938
N             :541.0000
```

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: 0x2b580a0>
act ~ sat + ibs + harv
<environment: 0x2b580a0>
ibs ~ sat + act + harv
<environment: 0x2b580a0>
harv ~ sat + act + ibs
<environment: 0x2b580a0>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998165 0.8645818 0.1963536 0.9998211
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
5450.424055    7.384532    1.244328 5590.687085
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.37 0.39 1.00
act  0.37 1.00 0.34 0.39
ibs  0.39 0.34 1.00 0.39
harv 1.00 0.39 0.39 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-30

	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)	2369.017	13562.51*	12663.453*	3363.013	6763.399*	4675.103
	(2417.79)	(1013.076)	(2350.168)	(2482.578)	(3030.263)	(2834.434)
SAT	11.268*	.	.	.	-30.4	9.077*
	(1.496)				(114.677)	(1.678)
ACT	.	312.227*	.	.	240.056	250.267*
		(43.919)			(128.782)	(50.041)
Iowa BS	.	.	77.421*	.	-51.931	-42.907
			(23.081)		(27.618)	(26.315)
Harvard SS	.	.	.	10.516*	38.353	.
				(1.512)	(114.645)	
N	493	512	529	487	438	477
RMSE	5301.243	5289.04	5510.911	5262.226	5112.412	5152.758
R^2	0.104	0.09	0.021	0.091	0.141	0.148
adj R^2	0.102	0.088	0.019	0.089	0.133	0.142

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    435 1.1413e+10
2    433 1.1317e+10  2  95298044 1.8231 0.1628
```

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)	2678.079	13286.256*	13978.603*	4003.056	6763.399*	4675.103
	(2441.234)	(1058.388)	(2528.219)	(2586.145)	(3030.263)	(2834.434)
SAT	11.11*	.	.	.	-30.4	9.077*
	(1.51)				(114.677)	(1.678)
ACT	.	323.571*	.	.	240.056	250.267*
		(45.864)			(128.782)	(50.041)
Iowa BS	.	.	64.906*	.	-51.931	-42.907
			(24.829)		(27.618)	(26.315)
Harvard SS	.	.	.	10.164*	38.353	.
				(1.575)	(114.645)	
N	477	477	477	438	438	477
RMSE	5276.631	5298.597	5529.659	5250.879	5112.412	5152.758
R^2	0.102	0.095	0.014	0.087	0.141	0.148
adj R^2	0.1	0.093	0.012	0.085	0.133	0.142

$*p \leq 0.05$

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   0.24138603
act   0.22411010
ibs  -0.07476221
```

```
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  0.052740418
act  0.045076405
ibs  0.004791213
```

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%    41.63   50.04   44.55
50%    52.62   59.81   55.02
75%    64.29   68.02   66.55
100%  100.00  100.00  100.00
```

```
mean    52.79    59.38    55.16
sd      17.02    14.60    16.03
var    289.70   213.00   257.00
NA's     0.00     0.00     0.00
N      477.00   477.00   477.00

$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
-13693.5  -3072.1    -79.9   3065.6  15734.4

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 13223.89    1127.14  11.732  < 2e-16 ***
satpoms        90.69      16.76   5.410 1.00e-07 ***
actpoms        77.86      15.57   5.001 8.04e-07 ***
ibspoms       -30.01      18.40  -1.631    0.104
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5153 on 473 degrees of freedom
Multiple R^2: 0.1476,   Adjusted R^2: 0.1422
F-statistic:  27.3 on 3 and 473 DF,  p-value: 2.678e-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  -0.01273831
act   0.08922306
ibs  -0.08999751
harv  0.01607471
```

```
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  0.0001394614
act  0.0068958101
ibs  0.0070170206
harv 0.0002221052
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-30

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	6969.591*	4619.648
	(3018.936)	(2824.834)
SAT	8.829*	9.066*
	(1.786)	(1.662)
ACT	272.331*	245.724*
	(53.119)	(49.336)
Iowa BS	-39.164	-40.583
	(28.063)	(26.036)
Major: Soc.	.	2160.984*
		(556.397)
Major: Nat.	.	4970.973*
		(575.887)
Prof. Parents: Yes	.	756.247
		(509.021)
Parent Network: Yes	.	1176.827*
		(501.397)
Gender: Male	.	-184.035
		(468.506)
N	489	489
RMSE	5545.38	5135.644
R^2	0.137	0.268
adj R^2	0.132	0.255

$*p \leq 0.05$

```
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 =  -420.580030671494 Denominator =  712.37874670845"
```

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

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

```
tval
```

```
difference
-0.5903882
```

```
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
 0.5552082
```

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
    -420.5800307       712.3787467        -0.5903882       480.0000000         0.5552082
```

```
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    485 14914353523
2    480 12659924736  5 2254428787 17.095 1.421e-15 ***
---
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

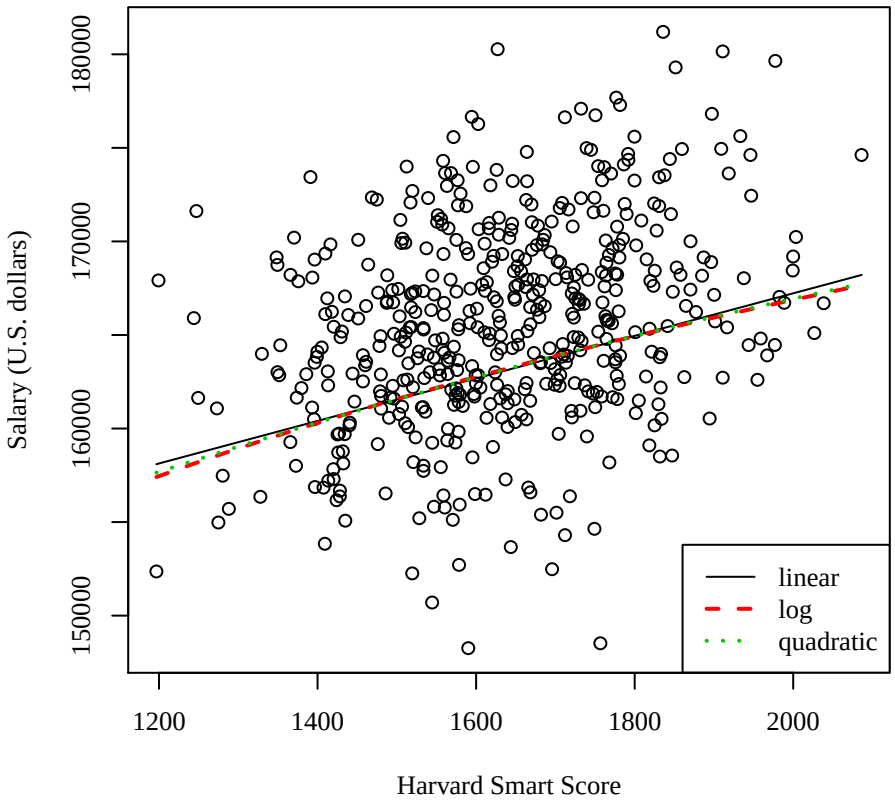
Table 4: Regression with sal3: Student-30

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	144348.479*	26585.893	137299.833*
	(2264.748)	(16293.179)	(17298.691)
Harvard SS	11.368*	.	20.053
	(1.36)		(21.174)
Gender: Male	141.837	141.08	142.27
	(427.327)	(427.26)	(427.692)
Major: Soc.	2493.106*	2501.162*	2500.231*
	(516.646)	(516.608)	(517.376)
Major: Nat.	4575.311*	4596.159*	4591.328*
	(519.874)	(519.922)	(521.773)
Prof. Parents: Yes	1625.116*	1615.872*	1615.814*
	(463.652)	(463.607)	(464.598)
Parent Network: Yes	-437.419	-451.421	-448.797
	(456.747)	(456.806)	(457.973)
ln(Harvard SS)	.	18438.566*	.
		(2202.848)	
Harvard SS^2	.	.	-0.003
			(0.006)
N	496	496	496
RMSE	4735.46	4734.711	4739.489
R^2	0.246	0.246	0.246
adj R^2	0.236	0.237	0.235

$*p \leq 0.05$

```
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%) 21151.06     H
S (30%) 23076.06     S
N (30%) 26184.04     N

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
H (40%) 21151.06      H
S (30%) 23076.06      S
N (30%) 26184.04      N

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

Table 5: Categorical Regressions: Student-30

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21151.058* (394.882)	23076.064* (423.2)	4619.648 (2824.834)	6780.632* (2845.266)
Major: Soc.	1925.007* (578.818)	.	2160.984* (556.397)	.
Major: Nat.	5032.985* (588.12)	.	4970.973* (575.887)	.
Major 2: Hum.	.	-1925.007* (578.818)	.	-2160.984* (556.397)
Major 2: Nat.	.	3107.979* (607.496)	.	2809.989* (590.133)
SAT	.	.	9.066* (1.662)	9.066* (1.662)
ACT	.	.	245.724* (49.336)	245.724* (49.336)
Iowa BS	.	.	-40.583 (26.036)	-40.583 (26.036)
Prof. Parents: Yes	.	.	756.247 (509.021)	756.247 (509.021)
Parent Network: Yes	.	.	1176.827* (501.397)	1176.827* (501.397)
Gender: Male	.	.	-184.035 (468.506)	-184.035 (468.506)
N	541	541	489	489
RMSE	5598.415	5598.415	5135.644	5135.644
R^2	0.12	0.12	0.268	0.268
adj R^2	0.117	0.117	0.255	0.255

$*p \leq 0.05$