Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 22
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%	7.64	1082.00	69.41	5875.00	8043.00	149100.00	1058.00
25%	18.49	1507.00	93.53	16650.00	19250.00	161200.00	1492.00
50%	21.54	1627.00	100.10	20710.00	23440.00	165600.00	1607.00
75%	24.87	1745.00	107.60	24420.00	27370.00	169200.00	1716.00
100%	34.41	2081.00	129.70	37330.00	42340.00	182400.00	2053.00
mean	21.73	1625.00	100.10	20570.00	23480.00	165300.00	1603.00
sd	4.65	166.20	10.00	5427.00	5808.00	6001.00	162.00
var	21.63	27620.00	99.98	29450000.00	33740000.00	36010000.00	26230.00
NA's	15.00	77.00	0.00	11.00	0.00	0.00	38.00
N	589.00	589.00	589.00	589.00	589.00	589.00	589.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                   major                    pnet
F             :308.0000   H             :204.0000   NO            :403.0000
M             :281.0000   S             :194.0000   YES           :186.0000
NA's          :  0.0000   N             :191.0000   NA's          :  0.0000
entropy       :  0.9985   NA's          :  0.0000   entropy       :  0.8997
normedEntropy:  0.9985   entropy       :  1.5844   normedEntropy:  0.8997
N             :589.0000   normedEntropy:  0.9996   N             :589.0000
                          N             :589.0000
          pprof
NO            :416.0000
YES           :173.0000
NA's          :  0.0000
entropy       :  0.8735
normedEntropy:  0.8735
N             :589.0000
```

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: 0x315a510>
act ~ sat + ibs + harv
<environment: 0x315a510>
ibs ~ sat + act + harv
<environment: 0x315a510>
harv ~ sat + act + ibs
<environment: 0x315a510>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998670 0.8509677 0.2218831 0.9998699
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
7517.993833    6.709956    1.285154 7686.353167
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.39 0.40 1.00
act  0.39 1.00 0.38 0.42
ibs  0.40 0.38 1.00 0.41
harv 1.00 0.42 0.41 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-22

	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)	2597.086	13573.576*	10579.577*	1757.482	1490.377	2105.609
	(2183.009)	(1054.067)	(2250.459)	(2228.764)	(2747.958)	(2613.332)
SAT	11.182*	.	.	.	59.499	8.234*
	(1.355)				(121.794)	(1.538)
ACT	.	321.268*	.	.	273.1*	218.105*
		(47.488)			(129.842)	(53.64)
Iowa BS	.	.	99.833*	.	13.933	4.582
			(22.378)		(26.278)	(24.999)
Harvard SS	.	.	.	11.551*	-51.526	.
				(1.365)	(121.727)	
N	541	563	578	503	458	526
RMSE	5084.037	5219.148	5340.265	5070.198	4976.54	5022.593
R^2	0.112	0.075	0.033	0.125	0.145	0.135
adj R^2	0.11	0.074	0.032	0.123	0.138	0.13

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    455 1.1230e+10
2    453 1.1219e+10  2  10924185 0.2205 0.8022
```

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)	3270.574	13276.948*	10562.579*	2974.88	1490.377	2105.609
	(2217.045)	(1080.363)	(2313.989)	(2300.024)	(2747.958)	(2613.332)
SAT	10.746*	.	.	.	59.499	8.234*
	(1.376)				(121.794)	(1.538)
ACT	.	332.583*	.	.	273.1*	218.105*
		(48.7)			(129.842)	(53.64)
Iowa BS	.	.	99.255*	.	13.933	4.582
			(23.013)		(26.278)	(24.999)
Harvard SS	.	.	.	10.76*	-51.526	.
				(1.41)	(121.727)	
N	526	526	526	458	458	526
RMSE	5101.535	5165.145	5296.91	5052.051	4976.54	5022.593
R^2	0.104	0.082	0.034	0.113	0.145	0.135
adj R^2	0.102	0.08	0.032	0.111	0.138	0.13

$*p \leq 0.05$

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   0.228133880
act   0.175215749
ibs   0.008022544
```

```
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 4.748887e-02
act 2.739607e-02
ibs 5.567396e-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%    40.61   44.44   43.62
50%    51.79   56.44   54.93
75%    63.68   70.36   65.84
100%  100.00  100.00  100.00
```

```
mean    52.51    56.39    54.73
sd      17.29    18.49    16.27
var    299.00   341.70   264.60
NA's     0.00     0.00     0.00
N      526.00   526.00   526.00

$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
-13434.9  -3388.1    -27.2   3686.9  15745.0

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 12804.97     922.02  13.888  < 2e-16 ***
satpoms        81.89      15.30   5.353 1.30e-07 ***
actpoms        58.39      14.36   4.066 5.52e-05 ***
ibspoms         2.49      13.58   0.183    0.855
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5023 on 522 degrees of freedom
Multiple R^2: 0.135, Adjusted R^2: 0.1301
F-statistic: 27.16 on 3 and 522 DF,  p-value: 2.448e-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   0.02294668
act   0.09834421
ibs   0.02490312
harv -0.01988379
```

```
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  0.0004502846
act  0.0083471272
ibs  0.0005303904
harv 0.0003380562
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-22

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	5553.469*	3156.35
	(2761.405)	(2593.124)
SAT	8.583*	8.237*
	(1.64)	(1.524)
ACT	202.468*	214.88*
	(56.823)	(52.834)
Iowa BS	-3.325	2.433
	(26.465)	(24.627)
Major: Soc.	.	1521.556*
		(524.36)
Major: Nat.	.	4880.365*
		(534.666)
Prof. Parents: Yes	.	919.027
		(476.533)
Parent Network: Yes	.	134.582
		(466.392)
Gender: Male	.	-532.781
		(435.565)
N	536	536
RMSE	5401.53	5014.171
R^2	0.115	0.245
adj R^2	0.11	0.233

$*p \leq 0.05$

```
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 =  784.4456243087 Denominator =  665.098686958466"
```

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

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

```
tval
```

```
difference
  1.179442
```

```
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
 0.2387542
```

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
     784.4456243       665.0986870         1.1794424       527.0000000         0.2387542
```

```
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    532 15521911665
2    527 13249786684  5 2272124981 18.074 < 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

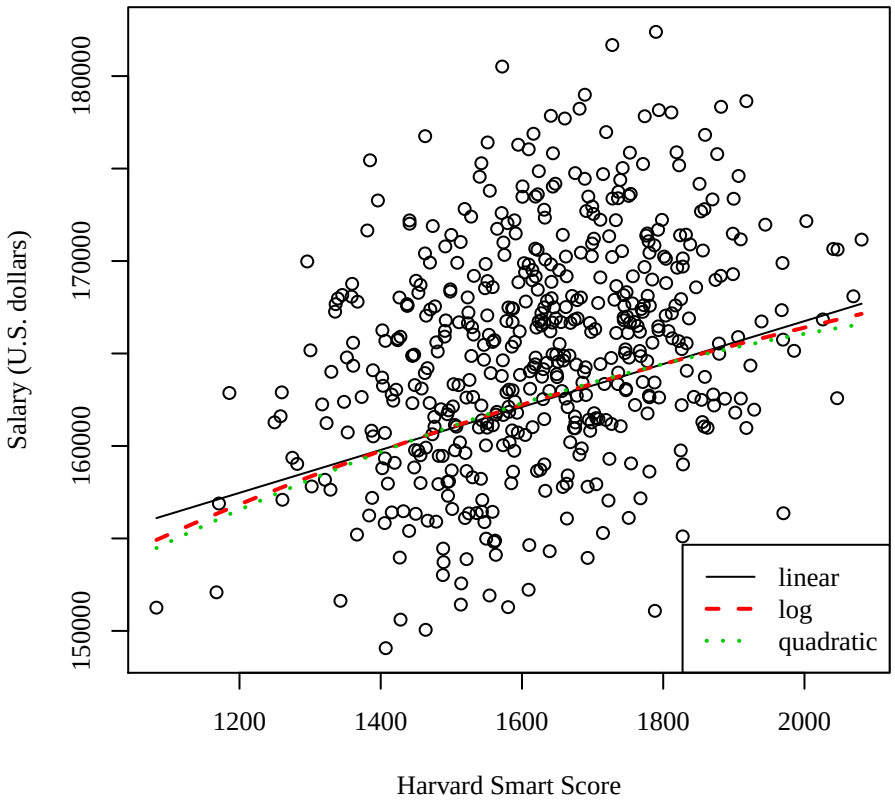
Table 4: Regression with sal3: Student-22

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	143537.501*	24244.284	127344.902*
	(2182.651)	(15648.949)	(15312.881)
Harvard SS	11.607*	.	31.804
	(1.324)		(18.952)
Gender: Male	-569.159	-565.282	-561.318
	(443.817)	(443.417)	(443.815)
Major: Soc.	2225.232*	2205.283*	2196.613*
	(537.003)	(536.602)	(537.596)
Major: Nat.	6680.145*	6657.771*	6643.992*
	(540.32)	(539.91)	(541.303)
Prof. Parents: Yes	917.017	903.809	897.032
	(481.508)	(481.136)	(481.804)
Parent Network: Yes	-226.527	-237.401	-242.424
	(473.234)	(472.829)	(473.402)
ln(Harvard SS)	.	18702.57*	.
		(2120.192)	
Harvard SS^2	.	.	-0.006
			(0.006)
N	512	512	512
RMSE	4968.226	4963.91	4967.531
R^2	0.325	0.326	0.327
adj R^2	0.317	0.318	0.317

$*p \leq 0.05$

```
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 (30%) 21370.10     H
S (30%) 22889.75     S
N (30%) 26345.83     N

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
H (30%) 21370.10      H
S (30%) 22889.75      S
N (30%) 26345.83      N

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

Table 5: Categorical Regressions: Student-22

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21370.098* (380.347)	22889.751* (390.027)	3156.35 (2593.124)	4677.906 (2615.46)
Major: Soc.	1519.653* (544.78)	.	1521.556* (524.36)	.
Major: Nat.	4975.734* (546.968)	.	4880.365* (534.666)	.
Major 2: Hum.	.	-1519.653* (544.78)	.	-1521.556* (524.36)
Major 2: Nat.	.	3456.08* (553.743)	.	3358.809* (539.013)
SAT	.	.	8.237* (1.524)	8.237* (1.524)
ACT	.	.	214.88* (52.834)	214.88* (52.834)
Iowa BS	.	.	2.433 (24.627)	2.433 (24.627)
Prof. Parents: Yes	.	.	919.027 (476.533)	919.027 (476.533)
Parent Network: Yes	.	.	134.582 (466.392)	134.582 (466.392)
Gender: Male	.	.	-532.781 (435.565)	-532.781 (435.565)
N	589	589	536	536
RMSE	5432.444	5432.444	5014.171	5014.171
R^2	0.128	0.128	0.245	0.245
adj R^2	0.125	0.125	0.233	0.233

$*p \leq 0.05$