Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 5
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%	5.64	994.90	64.18	5298.00	7868.00	148100.00	980.70
25%	18.66	1508.00	93.81	17050.00	19750.00	161200.00	1492.00
50%	22.17	1622.00	99.81	20640.00	23740.00	165200.00	1603.00
75%	25.20	1730.00	106.30	24440.00	27290.00	168900.00	1712.00
100%	35.27	2044.00	129.80	38940.00	44960.00	183200.00	2010.00
mean	22.07	1622.00	99.92	20610.00	23570.00	165200.00	1604.00
sd	4.90	154.70	9.68	5503.00	5861.00	5769.00	154.40
var	24.01	23930.00	93.76	30290000.00	34360000.00	33280000.00	23830.00
NA's	12.00	49.00	0.00	7.00	0.00	0.00	30.00
N	512.00	512.00	512.00	512.00	512.00	512.00	512.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                  major                    pnet                   pprof
F            :275.000   S            :188.0000   NO           :357.0000   NO           :346
    .0000
M            :237.000   H            :162.0000   YES          :155.0000   YES          :166
    .0000
NA's         :  0.000   N            :162.0000   NA's         :  0.0000   NA's         :  0
    .0000
entropy      :  0.996   NA's         :  0.0000   entropy      :  0.8846   entropy      :  0
    .9089
normedEntropy:  0.996   entropy      :  1.5813   normedEntropy:  0.8846   normedEntropy:  0
    .9089
N            :512.000   normedEntropy:  0.9977   N            :512.0000   N            :512
    .0000
                        N            :512.0000
```

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: 0x15b63d0>
act ~ sat + ibs + harv
<environment: 0x15b63d0>
ibs ~ sat + act + harv
<environment: 0x15b63d0>
harv ~ sat + act + ibs
<environment: 0x15b63d0>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998220 0.8542045 0.1709938 0.9998268
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
5619.084154    6.858922    1.206264 5774.253552
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.42 0.32 1.00
act  0.42 1.00 0.36 0.44
ibs  0.32 0.36 1.00 0.33
harv 1.00 0.44 0.33 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-5

	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)	115.315	12159.918*	8442.045*	576.652	-2227.489	-2968.657
	(2455.081)	(1081.236)	(2473.516)	(2533.855)	(3162.376)	(2963.848)
SAT	12.799*	.	.	.	69.764	9.601*
	(1.524)				(119.897)	(1.67)
ACT	.	383.548*	.	.	304.076*	246.7*
		(47.913)			(128.48)	(53.659)
Iowa BS	.	.	121.768*	.	27.85	27.82
			(24.635)		(27.273)	(26.006)
Harvard SS	.	.	.	12.336*	-60.591	.
				(1.556)	(119.74)	
N	475	493	505	457	416	463
RMSE	5106.082	5195.592	5379.824	5122.7	4950.146	4952.399
R^2	0.13	0.115	0.046	0.121	0.181	0.189
adj R^2	0.128	0.114	0.044	0.119	0.173	0.184

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    413 1.0102e+10
2    411 1.0071e+10  2  30631125 0.625 0.5358
```

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)	-791.57	12031.693*	8311.777*	-500.367	-2227.489	-2968.657
	(2471.089)	(1101.046)	(2563.029)	(2629.735)	(3162.376)	(2963.848)
SAT	13.367*	.	.	.	69.764	9.601*
	(1.534)				(119.897)	(1.67)
ACT	.	390.651*	.	.	304.076*	246.7*
		(48.8)			(128.48)	(53.659)
Iowa BS	.	.	123.341*	.	27.85	27.82
			(25.53)		(27.273)	(26.006)
Harvard SS	.	.	.	13.007*	-60.591	.
				(1.612)	(119.74)	
N	463	463	463	416	416	463
RMSE	5086.235	5143.067	5355.022	5065.258	4950.146	4952.399
R^2	0.141	0.122	0.048	0.136	0.181	0.189
adj R^2	0.139	0.12	0.046	0.134	0.173	0.184

$*p \leq 0.05$

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   0.25920420
act   0.20981923
ibs   0.04987066
```

```
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  0.058379535
act  0.037326295
ibs  0.002020891
```

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%    43.69   44.98   49.99
50%    55.52   54.22   60.76
75%    65.98   64.35   71.70
100%  100.00  100.00  100.00
```

```
mean    55.30    54.44    60.98
sd      16.55    14.87    15.11
var    273.80   221.00   228.30
NA's     0.00     0.00     0.00
N      463.00   463.00   463.00

$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
-13727  -3190     -58    3132   13518

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  9624.04    1150.84   8.363 7.41e-16 ***
satpoms        97.99      17.04   5.750 1.63e-08 ***
actpoms        73.10      15.90   4.598 5.53e-06 ***
ibspoms        18.26      17.07   1.070    0.285
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4952 on 459 degrees of freedom
Multiple R^2: 0.1895,   Adjusted R^2: 0.1842
F-statistic: 35.76 on 3 and 459 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 -1.00000000
sat   0.02868952
act   0.11595435
ibs   0.05030604
harv -0.02495249
```

```
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  0.0006749559
act  0.0111666865
ibs  0.0020787969
harv 0.0005104695
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-5

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	-1485.471	-2417.861
	(3186.528)	(2996.995)
SAT	10.373*	9.468*
	(1.793)	(1.662)
ACT	216.305*	244.782*
	(57.513)	(53.358)
Iowa BS	36.775	26.527
	(27.943)	(25.973)
Major: Soc.	.	1493.915*
		(557.479)
Major: Nat.	.	4772.301*
		(583.01)
Prof. Parents: Yes	.	985.166*
		(490.524)
Parent Network: Yes	.	808.731
		(504.944)
Gender: Male	.	338.964
		(460.644)
N	470	470
RMSE	5354.418	4952.305
R^2	0.172	0.299
adj R^2	0.166	0.287

$*p \leq 0.05$

```
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 =  176.4346929332 Denominator =  716.565460352268"
```

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

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

```
tval
```

```
difference
 0.2462227
```

```
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
 0.8056194
```

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
     176.4346929       716.5654604         0.2462227       461.0000000         0.8056194
```

```
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    466 13360123730
2    461 11306176353  5 2053947376 16.75 3.232e-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

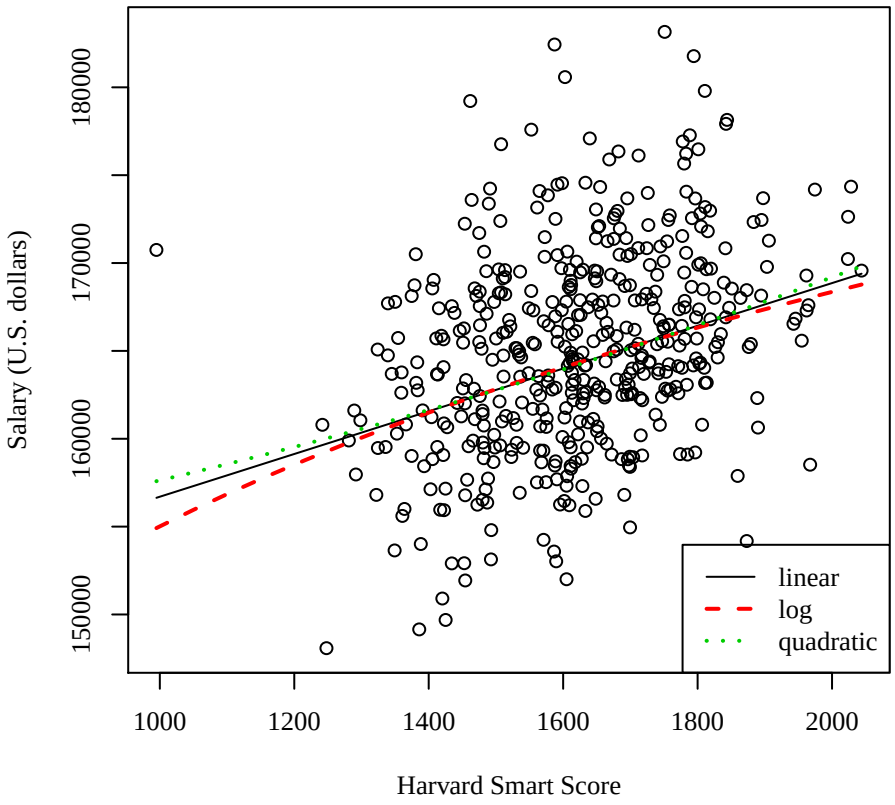
Table 4: Regression with sal3: Student-5

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	142351.787*	19877.191	149017.01*
	(2479.803)	(17816.83)	(17846.076)
Harvard SS	12.158*	.	3.844
	(1.51)		(22.097)
Gender: Male	956.685*	966.59*	952.975*
	(471.119)	(471.615)	(471.665)
Major: Soc.	2191.222*	2178.022*	2199.166*
	(562.776)	(563.412)	(563.699)
Major: Nat.	4701.854*	4709.597*	4700.06*
	(587.573)	(588.23)	(588.145)
Prof. Parents: Yes	1056.007*	1071.076*	1050.153*
	(503.649)	(504.136)	(504.362)
Parent Network: Yes	-28.884	-4.543	-44.087
	(508.826)	(509.473)	(510.898)
ln(Harvard SS)	.	19247.98*	.
		(2413.494)	
Harvard SS^2	.	.	0.003
			(0.007)
N	463	463	463
RMSE	5003.537	5009.323	5008.25
R^2	0.249	0.247	0.249
adj R^2	0.239	0.237	0.238

$*p \leq 0.05$

```
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%) 22771.95     S
H (30%) 21618.41     H
N (30%) 26454.61     N

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
S (40%) 22771.95      S
H (30%) 21618.41      H
N (30%) 26454.61      N

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

Table 5: Categorical Regressions: Student-5

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21618.413* (433.16)	22771.95* (402.093)	-2417.861 (2996.995)	-923.947 (2953.896)
Major: Soc.	1153.537 (591.022)	.	1493.915* (557.479)	.
Major: Nat.	4836.199* (612.581)	.	4772.301* (583.01)	.
Major 2: Hum.	.	-1153.537 (591.022)	.	-1493.915* (557.479)
Major 2: Nat.	.	3682.663* (591.022)	.	3278.387* (559.722)
SAT	.	.	9.468* (1.662)	9.468* (1.662)
ACT	.	.	244.782* (53.358)	244.782* (53.358)
Iowa BS	.	.	26.527 (25.973)	26.527 (25.973)
Prof. Parents: Yes	.	.	985.166* (490.524)	985.166* (490.524)
Parent Network: Yes	.	.	808.731 (504.944)	808.731 (504.944)
Gender: Male	.	.	338.964 (460.644)	338.964 (460.644)
N	512	512	470	470
RMSE	5513.225	5513.225	4952.305	4952.305
R^2	0.119	0.119	0.299	0.299
adj R^2	0.115	0.115	0.287	0.287

$*p \leq 0.05$