Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 34
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%	6.50	1198.00	74.92	1645.00	1419.00	148700.00	1179.00
25%	19.06	1509.00	93.79	16410.00	19050.00	161400.00	1489.00
50%	22.24	1619.00	100.10	20570.00	23390.00	165400.00	1600.00
75%	25.45	1733.00	106.40	24140.00	27250.00	169000.00	1715.00
100%	42.91	2110.00	136.20	37070.00	40920.00	182900.00	2079.00
mean	22.27	1623.00	100.20	20370.00	23300.00	165400.00	1603.00
sd	4.94	170.80	9.89	5417.00	5879.00	5535.00	169.70
var	24.44	29170.00	97.84	29350000.00	34570000.00	30630000.00	28790.00
NA's	18.00	65.00	0.00	5.00	0.00	0.00	24.00
N	528.00	528.00	528.00	528.00	528.00	528.00	528.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                  major                   pnet
F             :276.0000   S             :184.0000   NO            :367.0000
M             :252.0000   N             :176.0000   YES           :161.0000
NA's          :  0.0000   H             :168.0000   NA's          :  0.0000
entropy       :  0.9985   NA's          :  0.0000   entropy       :  0.8872
normedEntropy:  0.9985   entropy       :  1.5840   normedEntropy:  0.8872
N             :528.0000   normedEntropy:  0.9994   N             :528.0000
                          N             :528.0000
          pprof
NO            :390.0000
YES           :138.0000
NA's          :  0.0000
entropy       :  0.8288
normedEntropy:  0.8288
N             :528.0000
```

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: 0x23dd790>
act ~ sat + ibs + harv
<environment: 0x23dd790>
ibs ~ sat + act + harv
<environment: 0x23dd790>
harv ~ sat + act + ibs
<environment: 0x23dd790>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998704 0.8689245 0.2330867 0.9998733
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
7716.589422    7.629191    1.303928 7890.870020
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.38 0.40 1.00
act  0.38 1.00 0.41 0.41
ibs  0.40 0.41 1.00 0.40
harv 1.00 0.41 0.40 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-34

	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)	534.334	12946.557*	8312.096*	1051.385	573.315	-699.225
	(2116.871)	(1060.198)	(2347.03)	(2228.647)	(2840.938)	(2624.392)
SAT	12.397*	.	.	.	74.878	10.032*
	(1.313)				(123.763)	(1.477)
ACT	.	334.951*	.	.	264.715	196.819*
		(46.49)			(135.607)	(51.571)
Iowa BS	.	.	120.327*	.	5.031	6.86
			(23.314)		(27.377)	(25.54)
Harvard SS	.	.	.	11.887*	-65.563	.
				(1.366)	(123.953)	
N	499	506	523	459	421	483
RMSE	4984.751	5153.353	5288.896	5005.392	4932.529	4897.913
R^2	0.152	0.093	0.049	0.142	0.168	0.181
adj R^2	0.15	0.092	0.047	0.14	0.16	0.176

$*p \le 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    418 1.0129e+10
2    416 1.0121e+10  2   7675939 0.1577 0.8541
```

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)	694.972	12969.494*	9126.402*	1354.813	573.315	-699.225
	(2136.725)	(1097.026)	(2446.501)	(2307.43)	(2840.938)	(2624.392)
SAT	12.326*	.	.	.	74.878	10.032*
	(1.325)				(123.763)	(1.477)
ACT	.	336.219*	.	.	264.715	196.819*
		(48.112)			(135.607)	(51.571)
Iowa BS	.	.	113.021*	.	5.031	6.86
			(24.281)		(27.377)	(25.54)
Harvard SS	.	.	.	11.741*	-65.563	.
				(1.412)	(123.953)	
N	483	483	483	421	421	483
RMSE	4971.119	5144.903	5282.113	4992.948	4932.529	4897.913
R^2	0.152	0.092	0.043	0.142	0.168	0.181
adj R^2	0.151	0.09	0.041	0.14	0.16	0.176

$*p \leq 0.05$

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   0.29648425
act   0.17178566
ibs   0.01227108
```

```
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 0.0789630906
act 0.0249140804
ibs 0.0001233937
```

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%    34.66   30.86   34.49
50%    43.09   40.80   46.77
75%    51.96   51.69   59.38
100%  100.00  100.00  100.00
```

```
mean    43.33    41.40    47.15
sd      13.38    16.18    18.97
var    179.00   261.90   359.80
NA's     0.00     0.00     0.00
N      483.00   483.00   483.00

$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
-15702.5  -3398.9    204.6   3218.3  14140.7

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 12918.73     848.25  15.230  < 2e-16 ***
satpoms        90.37      13.30   6.794 3.24e-11 ***
actpoms        71.66      18.78   3.816 0.000153 ***
ibspoms         4.20      15.64   0.269 0.788364
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4898 on 479 degrees of freedom
Multiple R^2: 0.1807,   Adjusted R^2: 0.1755
F-statistic: 35.21 on 3 and 479 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   0.029650041
act   0.095273279
ibs   0.009010093
harv -0.025924312
```

```
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  0.00073190015
act  0.00761941447
ibs  0.00006753254
harv 0.00055940421
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-34

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	3277.802	-633.615
	(2908.648)	(2654.977)
SAT	9.528*	10.058*
	(1.636)	(1.474)
ACT	194.512*	211.228*
	(56.84)	(51.369)
Iowa BS	5.323	3.423
	(28.254)	(25.546)
Major: Soc.	.	2627.553*
		(549.99)
Major: Nat.	.	5635.52*
		(553.327)
Prof. Parents: Yes	.	1387.135*
		(507.314)
Parent Network: Yes	.	747.58
		(483.531)
Gender: Male	.	-1164.29*
		(448.283)
N	487	487
RMSE	5431.912	4884.783
R^2	0.141	0.313
adj R^2	0.136	0.301

$*p \leq 0.05$

```
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 =  639.554526223654 Denominator =  680.113982214268"
```

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

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

```
tval
```

```
difference
 0.9403637
```

```
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
  0.347506
```

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
      639.5545262       680.1139822         0.9403637      478.0000000        0.3475060
```

```
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    483 14251236265
2    478 11405606127  5 2845630138 23.852 < 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

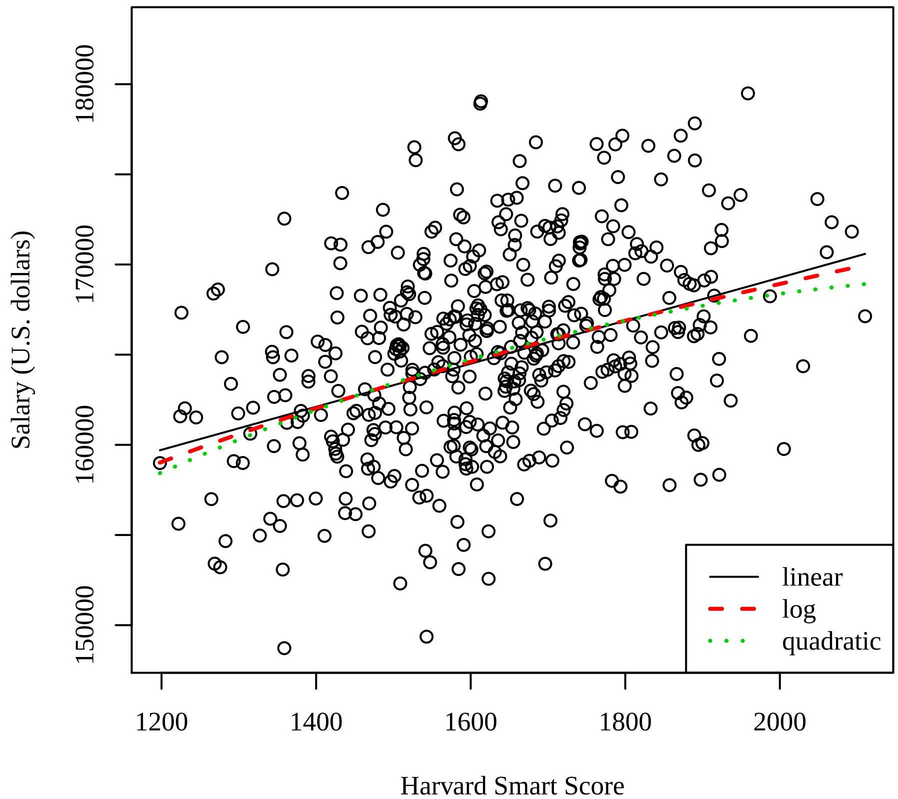
Table 4: Regression with sal3: Student-34

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	142987.323*	19580.822	121344.261*
	(2140.31)	(15294.774)	(14910.925)
Harvard SS	11.934*	.	38.793*
	(1.291)		(18.359)
Gender: Male	442.34	446.312	456.719
	(444.738)	(444.02)	(444.286)
Major: Soc.	2417.012*	2434.637*	2458.269*
	(540.009)	(539.131)	(540.062)
Major: Nat.	4389.703*	4385.516*	4377.47*
	(549.647)	(548.776)	(549.018)
Prof. Parents: Yes	1333.06*	1349.016*	1370.203*
	(509.647)	(508.846)	(509.635)
Parent Network: Yes	268.947	271.728	274.954
	(482.637)	(481.877)	(482.046)
ln(Harvard SS)	.	19327.709*	.
		(2071.279)	
Harvard SS^2	.	.	-0.008
			(0.006)
N	463	463	463
RMSE	4726.591	4719.262	4720.637
R^2	0.25	0.252	0.254
adj R^2	0.24	0.242	0.242

$*p \leq 0.05$

```
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 (30%) 23157.15     S
N (30%) 26021.16     N
H (30%) 20609.45     H

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

```
predictOMatic(cm2)
```

```
$major2
              fit major2
S (30%) 23157.15      S
N (30%) 26021.16      N
H (30%) 20609.45      H

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

Table 5: Categorical Regressions: Student-34

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	20609.454*	23157.154*	-633.615	1993.938
	(421.82)	(403.063)	(2654.977)	(2683.439)
Major: Soc.	2547.7*	.	2627.553*	.
	(583.431)		(549.99)	
Major: Nat.	5411.706*	.	5635.52*	.
	(589.726)		(553.327)	
Major 2: Hum.	.	-2547.7*	.	-2627.553*
		(583.431)		(549.99)
Major 2: Nat.	.	2864.006*	.	3007.968*
		(576.458)		(540.755)
SAT	.	.	10.058*	10.058*
			(1.474)	(1.474)
ACT	.	.	211.228*	211.228*
			(51.369)	(51.369)
Iowa BS	.	.	3.423	3.423
			(25.546)	(25.546)
Prof. Parents: Yes	.	.	1387.135*	1387.135*
			(507.314)	(507.314)
Parent Network: Yes	.	.	747.58	747.58
			(483.531)	(483.531)
Gender: Male	.	.	-1164.29*	-1164.29*
			(448.283)	(448.283)
N	528	528	487	487
RMSE	5467.413	5467.413	4884.783	4884.783
R^2	0.139	0.139	0.313	0.313
adj R^2	0.135	0.135	0.301	0.301

$*p \leq 0.05$