Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 35
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%	8.76	1201.00	71.92	2668.00	5235.00	145800.00	1187.00
25%	18.83	1527.00	93.31	16560.00	18960.00	161600.00	1510.00
50%	22.36	1624.00	99.91	20200.00	22820.00	165600.00	1603.00
75%	25.58	1733.00	106.70	24360.00	27320.00	168900.00	1718.00
100%	36.37	2128.00	129.70	35920.00	38790.00	185700.00	2098.00
mean	22.15	1631.00	100.00	20330.00	23150.00	165400.00	1610.00
sd	4.81	154.30	10.28	5612.00	5914.00	5597.00	151.50
var	23.13	23790.00	105.70	31500000.00	34980000.00	31330000.00	22950.00
NA's	13.00	42.00	0.00	18.00	0.00	0.00	26.00
N	579.00	579.00	579.00	579.00	579.00	579.00	579.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                   major                    pnet                    pprof
F             :323.0000   H             :207.0000   NO            :417.0000   NO            :382.000
M             :256.0000   S             :195.0000   YES           :162.0000   YES           :197.000
NA's          :  0.0000   N             :177.0000   NA's          :  0.0000   NA's          :  0.000
entropy       :  0.9903   NA's          :  0.0000   entropy       :  0.8552   entropy       :  0.925
normedEntropy:  0.9903   entropy       :  1.5820   normedEntropy:  0.8552   normedEntropy:  0.925
N             :579.0000   normedEntropy:  0.9981   N             :579.0000   N             :579.000
                          N             :579.0000
```

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: 0x1b1e4c0>
act ~ sat + ibs + harv
<environment: 0x1b1e4c0>
ibs ~ sat + act + harv
<environment: 0x1b1e4c0>
harv ~ sat + act + ibs
<environment: 0x1b1e4c0>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998197 0.8575360 0.1981906 0.9998241
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
5545.522454    7.019315    1.247179 5684.526786
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.38 0.40  1.0
act  0.38 1.00 0.34  0.4
ibs  0.40 0.34 1.00  0.4
harv 1.00 0.40 0.40  1.0
```

```
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-35

	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)	-6451.357*	11691.652*	9898.452*	-6527.522*	-6431.363*	-6166.538*
	(2324.758)	(1062.044)	(2282.837)	(2322.577)	(2757.553)	(2677.113)
SAT	16.648*	.	.	.	119.621	13.91*
	(1.437)				(109.778)	(1.625)
ACT	.	390.483*	.	.	340.954*	235.015*
		(46.945)			(121.1)	(49.693)
Iowa BS	.	.	104.41*	.	-6.194	-10.704
			(22.719)		(24.637)	(23.582)
Harvard SS	.	.	.	16.445*	-105.822	.
				(1.418)	(109.828)	
N	536	548	561	519	484	524
RMSE	5051.56	5301.091	5514.06	4996.949	4944.397	4969.489
R^2	0.201	0.112	0.036	0.206	0.238	0.231
adj R^2	0.199	0.111	0.035	0.205	0.231	0.227

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    481 1.1734e+10
2    479 1.1710e+10  2  24033962 0.4916  0.612
```

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)	-6299.778*	11596.791*	9731.691*	-6638.703*	-6431.363*	-6166.538*
	(2360.298)	(1082.192)	(2374.539)	(2428.444)	(2757.553)	(2677.113)
SAT	16.56*	.	.	.	119.621	13.91*
	(1.46)				(109.778)	(1.625)
ACT	.	395.839*	.	.	340.954*	235.015*
		(47.77)			(121.1)	(49.693)
Iowa BS	.	.	106.573*	.	-6.194	-10.704
			(23.696)		(24.637)	(23.582)
Harvard SS	.	.	.	16.535*	-105.822	.
				(1.483)	(109.828)	
N	524	524	524	484	484	524
RMSE	5067.283	5318.427	5550.898	5033.085	4944.397	4969.489
R^2	0.198	0.116	0.037	0.205	0.238	0.231
adj R^2	0.196	0.115	0.035	0.203	0.231	0.227

$*p \leq 0.05$

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   0.35147947
act   0.20307363
ibs  -0.01990191
```

```
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 0.1083392462
act 0.0330609825
ibs 0.0003045655
```

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%    36.00   36.17   35.50
50%    49.44   48.08   45.66
75%    61.04   59.66   57.79
100%  100.00  100.00  100.00
```

```
mean    48.41    48.05    46.38
sd      17.63    17.73    16.67
var    310.90   314.30   277.80
NA's     0.00     0.00     0.00
N      524.00   524.00   524.00

$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
-14620.0  -3338.7   -269.1   3236.3  13724.9

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 11636.666    812.696  14.319  < 2e-16 ***
satpoms       126.660     14.795   8.561  < 2e-16 ***
actpoms        64.888     13.720   4.729 2.91e-06 ***
ibspoms        -6.185     13.626  -0.454     0.65
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4969 on 520 degrees of freedom
Multiple R^2: 0.2314,   Adjusted R^2: 0.2269
F-statistic: 52.18 on 3 and 520 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   0.04972669
act   0.12759094
ibs  -0.01148641
harv -0.04398166
```

```
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  0.0018898313
act  0.0126164334
ibs  0.0001005992
harv 0.0014775866
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-35

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	-3198.674	-5431.5*
	(2828.588)	(2647.541)
SAT	13.456*	13.554*
	(1.724)	(1.605)
ACT	225.55*	236.661*
	(52.853)	(49.272)
Iowa BS	-2.916	-9.543
	(24.917)	(23.223)
Major: Soc.	.	706.719
		(513.866)
Major: Nat.	.	4538.271*
		(525.99)
Prof. Parents: Yes	.	598.068
		(452.332)
Parent Network: Yes	.	1408.785*
		(475.645)
Gender: Male	.	593.316
		(434.38)
N	541	541
RMSE	5334.479	4946.176
R^2	0.199	0.317
adj R^2	0.194	0.307

$*p \leq 0.05$

```
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 =  -810.716650487932 Denominator =  679.40630807822"
```

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

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

```
tval
```

```
difference
 -1.193272
```

```
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
 0.2332947
```

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
    -810.7166505        679.4063081       -1.1932722       532.0000000        0.2332947
```

```
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    537 15281229408
2    532 13015199312  5 2266030097 18.525 < 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

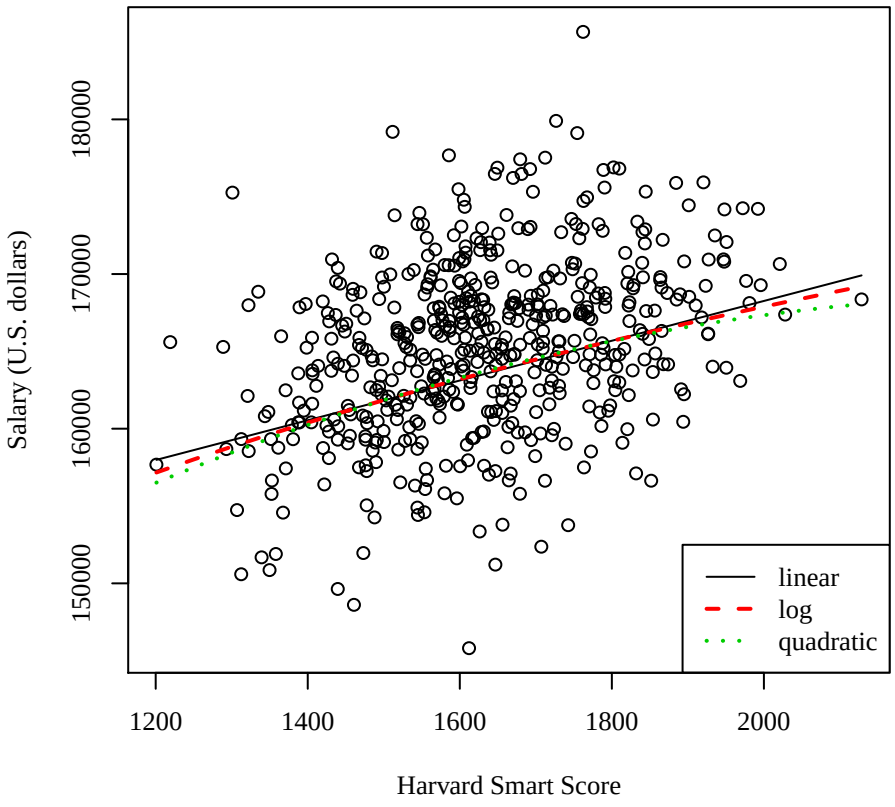
Table 4: Regression with sal3: Student-35

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	142540.328*	8130.143	119569.206*
	(2262.932)	(16361.029)	(18019.893)
Harvard SS	12.859*	.	41.099
	(1.364)		(22.02)
Gender: Male	-463.494	-450.236	-432.232
	(427.257)	(426.729)	(427.688)
Major: Soc.	1592.97*	1602.59*	1611.669*
	(506.879)	(506.339)	(506.777)
Major: Nat.	4424.82*	4419.144*	4412.922*
	(517.099)	(516.531)	(516.865)
Prof. Parents: Yes	478.274	476.021	469.809
	(442.248)	(441.765)	(442.026)
Parent Network: Yes	-102.909	-107.935	-108.046
	(465.709)	(465.207)	(465.44)
ln(Harvard SS)	.	21018.577*	.
		(2212.931)	
Harvard SS^2	.	.	-0.009
			(0.007)
N	537	537	537
RMSE	4866.491	4861.205	4863.505
R^2	0.235	0.236	0.237
adj R^2	0.226	0.228	0.227

$*p \leq 0.05$

```
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%) 21528.76     H
S (30%) 22316.37     S
N (30%) 25965.20     N

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

```
predictOMatic(cm2)
```

```
$major2
            fit major2
H (40%) 21528.76      H
S (30%) 22316.37      S
N (30%) 25965.20      N

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

Table 5: Categorical Regressions: Student-35

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21528.759* (390.006)	22316.372* (401.827)	-5431.5* (2647.541)	-4724.782 (2652.578)
Major: Soc.	787.612 (559.973)	.	706.719 (513.866)	.
Major: Nat.	4436.437* (574.448)	.	4538.271* (525.99)	.
Major 2: Hum.	.	-787.612 (559.973)	.	-706.719 (513.866)
Major 2: Nat.	.	3648.825* (582.538)	.	3831.552* (534.107)
SAT	.	.	13.554* (1.605)	13.554* (1.605)
ACT	.	.	236.661* (49.272)	236.661* (49.272)
Iowa BS	.	.	-9.543 (23.223)	-9.543 (23.223)
Prof. Parents: Yes	.	.	598.068 (452.332)	598.068 (452.332)
Parent Network: Yes	.	.	1408.785* (475.645)	1408.785* (475.645)
Gender: Male	.	.	593.316 (434.38)	593.316 (434.38)
N	579	579	541	541
RMSE	5611.212	5611.212	4946.176	4946.176
R^2	0.103	0.103	0.317	0.317
adj R^2	0.1	0.1	0.307	0.307

$*p \leq 0.05$