Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 44
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%	9.04	1157.00	62.67	-476.50	3460.00	148500.00	1134.00
25%	18.13	1496.00	91.95	16670.00	19850.00	161600.00	1483.00
50%	21.64	1618.00	98.44	20390.00	23300.00	165500.00	1599.00
75%	25.52	1738.00	105.20	24050.00	27190.00	169000.00	1710.00
100%	35.63	2066.00	129.20	34590.00	38610.00	182000.00	2124.00
mean	21.69	1614.00	98.78	20380.00	23430.00	165500.00	1596.00
sd	5.05	167.40	10.09	5348.00	5736.00	5691.00	165.50
var	25.52	28020.00	101.80	28600000.00	32900000.00	32390000.00	27380.00
NA's	16.00	52.00	0.00	9.00	0.00	0.00	30.00
N	527.00	527.00	527.00	527.00	527.00	527.00	527.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                    major                     pnet
F             :278.0000   N             :184.0000   NO            :359.0000
M             :249.0000   S             :175.0000   YES           :168.0000
NA's          :  0.0000   H             :168.0000   NA's          :  0.0000
entropy       :  0.9978   NA's          :  0.0000   entropy       :  0.9031
normedEntropy:  0.9978   entropy       :  1.5840   normedEntropy:  0.9031
N             :527.0000   normedEntropy:  0.9994   N             :527.0000
                          N             :527.0000
          pprof
NO            :377.0000
YES           :150.0000
NA's          :  0.0000
entropy       :  0.8617
normedEntropy:  0.8617
N             :527.0000
```

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: 0x1f39f98>
act ~ sat + ibs + harv
<environment: 0x1f39f98>
ibs ~ sat + act + harv
<environment: 0x1f39f98>
harv ~ sat + act + ibs
<environment: 0x1f39f98>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998367 0.8560008 0.2858186 0.9998407
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
6122.001416    6.944484    1.400204 6276.655978
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.41 0.47 1.00
act  0.41 1.00 0.43 0.44
ibs  0.47 0.43 1.00 0.47
harv 1.00 0.44 0.47 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-44

	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)	2898.957	13966.844*	10028.769*	1791.247	2269.922	2288.648
	(2195.767)	(1023.459)	(2270.424)	(2275.082)	(2835.233)	(2662.559)
SAT	10.947*	.	.	.	53.614	8.42*
	(1.369)				(115.598)	(1.617)
ACT	.	296.315*	.	.	199.284	161.485*
		(46.033)			(129.796)	(53.096)
Iowa BS	.	.	104.744*	.	-1.451	11.578
			(22.852)		(29.206)	(27.315)
Harvard SS	.	.	.	11.502*	-44.28	.
				(1.402)	(115.567)	
N	488	502	518	467	427	473
RMSE	5006.634	5175.403	5247.641	5069.7	5075.599	4998.449
R^2	0.116	0.077	0.039	0.126	0.135	0.133
adj R^2	0.114	0.075	0.037	0.125	0.127	0.127

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    424 1.0875e+10
2    422 1.0871e+10  2   3815003 0.074 0.9286
```

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)	3193.024	14119.575*	9292.204*	2336.141	2269.922	2288.648
	(2239.692)	(1055.235)	(2403.135)	(2379.722)	(2835.233)	(2662.559)
SAT	10.767*	.	.	.	53.614	8.42*
	(1.396)				(115.598)	(1.617)
ACT	.	288.254*	.	.	199.284	161.485*
		(47.382)			(129.796)	(53.096)
Iowa BS	.	.	112.048*	.	-1.451	11.578
			(24.173)		(29.206)	(27.315)
Harvard SS	.	.	.	11.163*	-44.28	.
				(1.466)	(115.567)	
N	473	473	473	427	427	473
RMSE	5047.524	5158	5238.676	5100.675	5075.599	4998.449
R^2	0.112	0.073	0.044	0.12	0.135	0.133
adj R^2	0.11	0.071	0.042	0.118	0.127	0.127

$*p \leq 0.05$

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   0.23378553
act   0.13907263
ibs   0.01956859
```

```
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 0.0501248104
act 0.0170990873
ibs 0.0003321182
```

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%    34.56   44.98   34.90
50%    47.57   54.55   47.07
75%    62.20   64.60   58.16
100%  100.00  100.00  100.00
```

```
mean    47.62    55.01    46.65
sd      18.84    15.14    16.83
var    355.10   229.30   283.10
NA's     0.00     0.00     0.00
N      473.00   473.00   473.00

$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
-20598.0  -3259.0    -30.6   3327.7  13562.2

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 14025.402    927.662  15.119  < 2e-16 ***
satpoms        83.289     15.995   5.207 2.87e-07 ***
actpoms        42.939     14.118   3.041  0.00249 **
ibspoms         7.628     17.995   0.424  0.67186
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4998 on 469 degrees of freedom
Multiple R^2: 0.133, Adjusted R^2: 0.1275
F-statistic: 23.99 on 3 and 469 DF,  p-value: 1.857e-14
```

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   0.022571430
act   0.074532479
ibs  -0.002418876
harv -0.018648392
```

```
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  0.000440997188
act  0.004832888260
ibs  0.000005062046
harv 0.000300974829
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-44

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	7184.705*	2481.974
	(2866.999)	(2672.709)
SAT	7.658*	8.334*
	(1.74)	(1.59)
ACT	215.199*	161.115*
	(56.642)	(52.228)
Iowa BS	-6.941	14.823
	(29.328)	(26.877)
Major: Soc.	.	724.869
		(559.623)
Major: Nat.	.	4704.183*
		(553.851)
Prof. Parents: Yes	.	1837.557*
		(504.07)
Parent Network: Yes	.	1510.813*
		(484.38)
Gender: Male	.	-473.714
		(452.42)
N	482	482
RMSE	5425.084	4944.101
R^2	0.115	0.273
adj R^2	0.11	0.26

$*p \leq 0.05$

```
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 =  326.744231060333 Denominator =  720.385011113422"
```

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

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

```
tval
```

```
difference
 0.4535689
```

```
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
 0.6503472
```

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
     326.7442311       720.3850111         0.4535689       473.0000000         0.6503472
```

```
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    478 14068276732
2    473 11562075187  5 2506201545 20.506 < 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

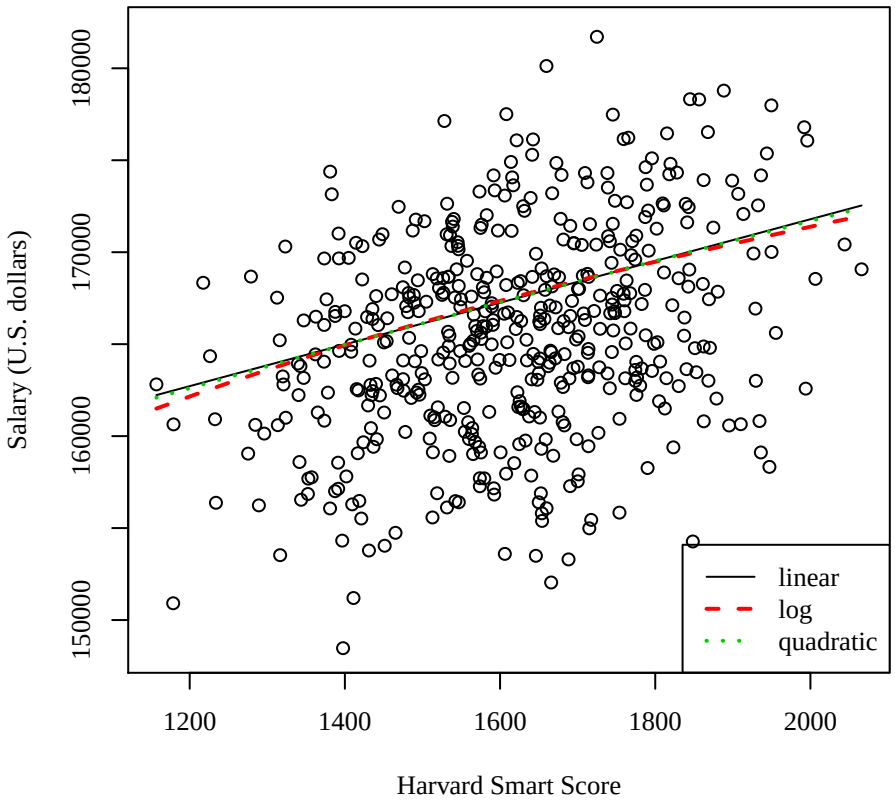
Table 4: Regression with sal3: Student-44

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	143900.683*	28832.298	141920.73*
	(2187.625)	(15524.61)	(15874.006)
Harvard SS	11.342*	.	13.825
	(1.319)		(19.756)
Gender: Male	504.424	505.934	504.873
	(441.798)	(441.793)	(442.277)
Major: Soc.	1874.177*	1876.166*	1874.316*
	(545.208)	(545.222)	(545.784)
Major: Nat.	5206.875*	5205.449*	5206.009*
	(542.417)	(542.412)	(543.032)
Prof. Parents: Yes	1588.625*	1580.591*	1586.734*
	(487.834)	(487.859)	(488.579)
Parent Network: Yes	606.021	613.045	607.793
	(476.221)	(476.244)	(476.931)
ln(Harvard SS)	.	18069.858*	.
		(2101.329)	
Harvard SS^2	.	.	-0.001
			(0.006)
N	475	475	475
RMSE	4787.176	4787.231	4792.218
R^2	0.27	0.27	0.27
adj R^2	0.26	0.26	0.259

$*p \leq 0.05$

```
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
N (30%) 25939.99     N
S (30%) 22711.76     S
H (30%) 21429.76     H

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

```
predictOMatic(cm2)
```

```
$major2
             fit major2
N (30%) 25939.99      N
S (30%) 22711.76      S
H (30%) 21429.76      H

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

Table 5: Categorical Regressions: Student-44

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21429.756* (418.058)	22711.762* (409.612)	2481.974 (2672.709)	3206.843 (2668.247)
Major: Soc.	1282.006* (585.281)	.	724.869 (559.623)	.
Major: Nat.	4510.229* (578.228)	.	4704.183* (553.851)	.
Major 2: Hum.	.	-1282.006* (585.281)	.	-724.869 (559.623)
Major 2: Nat.	.	3228.224* (572.151)	.	3979.314* (555.555)
SAT	.	.	8.334* (1.59)	8.334* (1.59)
ACT	.	.	161.115* (52.228)	161.115* (52.228)
Iowa BS	.	.	14.823 (26.877)	14.823 (26.877)
Prof. Parents: Yes	.	.	1837.557* (504.07)	1837.557* (504.07)
Parent Network: Yes	.	.	1510.813* (484.38)	1510.813* (484.38)
Gender: Male	.	.	-473.714 (452.42)	-473.714 (452.42)
N	527	527	482	482
RMSE	5418.652	5418.652	4944.101	4944.101
R^2	0.111	0.111	0.273	0.273
adj R^2	0.108	0.108	0.26	0.26

$*p \leq 0.05$