Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 16
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%	7.98	1196.00	71.70	5153.00	7333.00	150300.00	1184.00
25%	18.83	1517.00	93.76	16200.00	19210.00	161600.00	1502.00
50%	21.92	1617.00	100.20	20300.00	23350.00	165500.00	1603.00
75%	25.54	1723.00	107.00	23710.00	27220.00	169300.00	1704.00
100%	38.05	2109.00	132.50	36970.00	41370.00	182500.00	2086.00
mean	22.08	1618.00	100.10	20280.00	23380.00	165400.00	1600.00
sd	5.04	160.00	10.05	5537.00	5828.00	5547.00	158.80
var	25.36	25590.00	100.90	30660000.00	33970000.00	30760000.00	25210.00
NA’s	26.00	55.00	0.00	9.00	0.00	0.00	24.00
N	572.00	572.00	572.00	572.00	572.00	572.00	572.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

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

```
          gender                  major                  pnet
F             :293.0000  S             :206.0000  NO            :380.0000
M             :279.0000  N             :197.0000  YES           :192.0000
NA's          :  0.0000  H             :169.0000  NA's          :  0.0000
entropy       :  0.9996  NA's          :  0.0000  entropy       :  0.9206
normedEntropy:  0.9996   entropy       :  1.5800  normedEntropy:  0.9206
N             :572.0000  normedEntropy:  0.9968   N             :572.0000
                         N             :572.0000
          pprof
NO            :404.0000
YES           :168.0000
NA's          :  0.0000
entropy       :  0.8735
normedEntropy:  0.8735
N             :572.0000
```

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: 0x2347cf0>
act ~ sat + ibs + harv
<environment: 0x2347cf0>
ibs ~ sat + act + harv
<environment: 0x2347cf0>
harv ~ sat + act + ibs
<environment: 0x2347cf0>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998293 0.8524074 0.2033692 0.9998339
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
5858.199147    6.775407    1.255287 6021.688708
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.43 0.37 1.00
act  0.43 1.00 0.38 0.45
ibs  0.37 0.38 1.00 0.38
harv 1.00 0.45 0.38 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-16

	SAT	ACT	IBS	Harvard SS	All	Best
	Estimate	Estimate	Estimate	Estimate	Estimate	Estimate
	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)
(Intercept)	297.915	13908.118*	10982.386*	-939.953	-2.038	1032.149
	(2270.919)	(1043.15)	(2313.762)	(2296.075)	(2955.049)	(2804.519)
SAT	12.534*	.	.	.	-24.218	10.569*
	(1.414)				(115.503)	(1.659)
ACT	.	288.772*	.	.	104.952	142.084*
		(46.104)			(125.121)	(51.787)
Iowa BS	.	.	92.963*	.	0.902	-7.545
			(23.022)		(26.965)	(25.499)
Harvard SS	.	.	.	13.053*	34.857	.
				(1.414)	(115.43)	
N	539	537	563	509	466	516
RMSE	5200.45	5390.54	5462.839	5084.189	5153.392	5220.57
R^2	0.128	0.068	0.028	0.144	0.142	0.136
adj R^2	0.126	0.067	0.027	0.142	0.135	0.131

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    463 1.2245e+10
2    461 1.2243e+10  2   2506432 0.0472 0.9539
```

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	ACT	IBS	Harvard SS	All	Best
	Estimate	Estimate	Estimate	Estimate	Estimate	Estimate
	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)
(Intercept)	570.299	14192.296*	11988.725*	-43.965	-2.038	1032.149
	(2341.291)	(1060.03)	(2422.94)	(2427.758)	(2955.049)	(2804.519)
SAT	12.349*	.	.	.	-24.218	10.569*
	(1.457)				(115.503)	(1.659)
ACT	.	277.143*	.	.	104.952	142.084*
		(46.76)			(125.121)	(51.787)
Iowa BS	.	.	83.218*	.	0.902	-7.545
			(24.097)		(26.965)	(25.499)
Harvard SS	.	.	.	12.49*	34.857	.
				(1.492)	(115.43)	
N	516	516	516	466	466	516
RMSE	5249.664	5422.165	5540.481	5168.799	5153.392	5220.57
R^2	0.123	0.064	0.023	0.131	0.142	0.136
adj R^2	0.121	0.062	0.021	0.129	0.135	0.131

$*p \leq 0.05$

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    0.27097073
act    0.12036918
ibs   -0.01307586
```

```
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 0.0684936705
act 0.0127073567
ibs 0.0001478089
```

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%    36.07    36.46    35.44
50%    46.46    46.83    45.91
75%    58.38    57.80    57.68
100%  100.00   100.00   100.00
```

```
mean   46.92  46.58  45.98
sd     16.99  16.65  17.60
var   288.80 277.30 309.60
NA's    0.00   0.00   0.00
N     516.00 516.00 516.00

$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
-13796.2  -3881.5     11.9   3521.3  15758.2

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 14138.80     842.68  16.778  < 2e-16 ***
satpoms        95.35      14.97   6.370 4.22e-10 ***
actpoms        42.73      15.57   2.744  0.00629 **
ibspoms        -4.59      15.51  -0.296  0.76743
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5221 on 512 degrees of freedom
Multiple R^2: 0.1357,   Adjusted R^2: 0.1306
F-statistic: 26.79 on 3 and 512 DF,  p-value: 4.172e-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  -0.009765268
act   0.039037359
ibs   0.001557355
harv  0.014063186
```

```
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  0.000081829373
act  0.001309551822
ibs  0.000002081017
harv 0.000169727747
```

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

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-16

	Test Scores Only	All Predictors
	Estimate	Estimate
	(S.E.)	(S.E.)
(Intercept)	4606.677	1039.785
	(2916.519)	(2831.06)
SAT	9.948*	10.527*
	(1.731)	(1.649)
ACT	161.068*	150.187*
	(54.277)	(51.541)
Iowa BS	-6.207	-8.3
	(26.527)	(25.18)
Major: Soc.	.	1622.964*
		(573.832)
Major: Nat.	.	3998.092*
		(569.465)
Prof. Parents: Yes	.	1578.587*
		(500.167)
Parent Network: Yes	.	1163.197*
		(483.71)
Gender: Male	.	467.49
		(457.239)
N	525	525
RMSE	5487.645	5205.766
R^2	0.123	0.218
adj R^2	0.117	0.206

$*p \leq 0.05$

```
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 =  415.390461038185 Denominator =  671.493395522024"
```

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

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

```
tval
```

```
difference
 0.6186069
```

```
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
 0.5364483
```

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
      415.3904610       671.4933955         0.6186069       516.0000000         0.5364483
```

```
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    521 15689524705
2    516 13983601734  5 1705922970 12.59 1.494e-11 ***
---
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

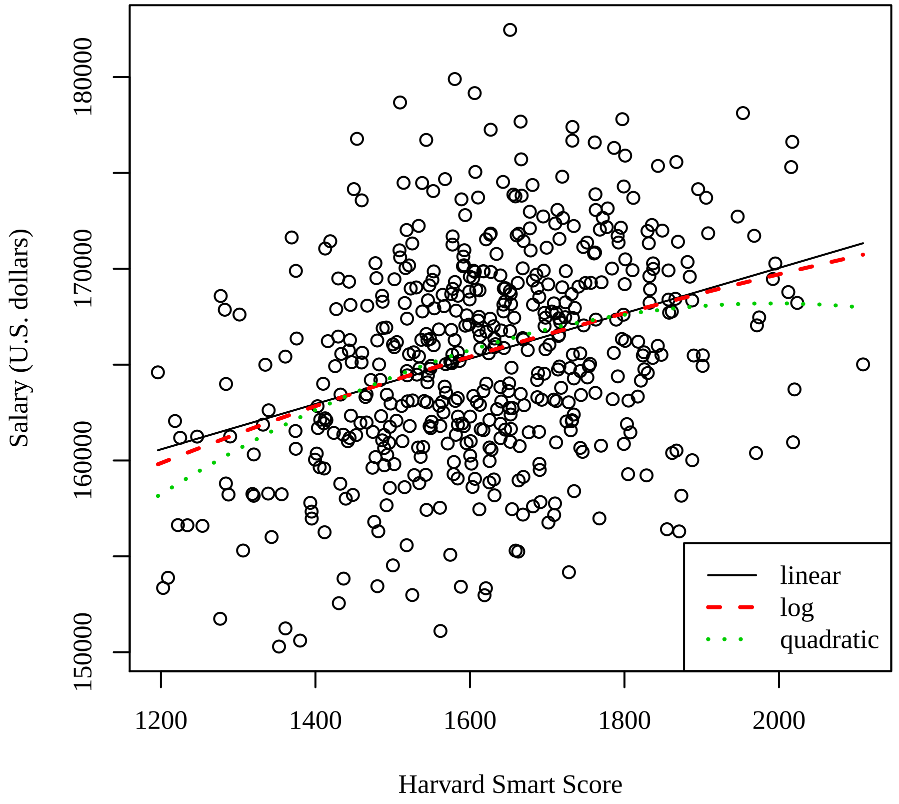
Table 4: Regression with sal3: Student-16

	Linear	Log	Quadratic
	Estimate	Estimate	Estimate
	(S.E.)	(S.E.)	(S.E.)
(Intercept)	143939.137*	20705.398	102822.628*
	(2218.878)	(15641.362)	(15183.451)
Harvard SS	11.831*	.	63.143*
	(1.327)		(18.794)
Gender: Male	-361.21	-371.779	-411.79
	(424.389)	(422.999)	(422.118)
Major: Soc.	2443.734*	2445.825*	2472.32*
	(527.538)	(525.911)	(524.317)
Major: Nat.	4998.739*	4993.098*	4978.268*
	(531.394)	(529.775)	(528.097)
Prof. Parents: Yes	429.968	455.959	537.447
	(463.393)	(462.029)	(462.143)
Parent Network: Yes	-663.392	-682.768	-753.76
	(449.599)	(448.148)	(447.983)
ln(Harvard SS)	.	19282.468*	.
		(2115.565)	
Harvard SS^2	.	.	-0.016*
			(0.006)
N	517	517	517
RMSE	4810.075	4795.359	4779.753
R^2	0.258	0.262	0.268
adj R^2	0.249	0.253	0.258

$*p \leq 0.05$

```
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%) 23036.23     S
N (30%) 25453.70     N
H (30%) 21387.06     H

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

```
predictOMatic(cm2)
```

```
$major2
             fit  major2
S (40%) 23036.23      S
N (30%) 25453.70      N
H (30%) 21387.06      H

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

Table 5: Categorical Regressions: Student-16

	major	major2	major full	major2 full
	Estimate	Estimate	Estimate	Estimate
	(S.E.)	(S.E.)	(S.E.)	(S.E.)
(Intercept)	21387.062*	23036.225*	1039.785	2662.749
	(430.868)	(390.26)	(2831.06)	(2836.661)
Major: Soc.	1649.163*	.	1622.964*	.
	(581.335)		(573.832)	
Major: Nat.	4066.637*	.	3998.092*	.
	(587.289)		(569.465)	
Major 2: Hum.	.	-1649.163*	.	-1622.964*
		(581.335)		(573.832)
Major 2: Nat.	.	2417.474*	.	2375.128*
		(558.179)		(540.738)
SAT	.	.	10.527*	10.527*
			(1.649)	(1.649)
ACT	.	.	150.187*	150.187*
			(51.541)	(51.541)
Iowa BS	.	.	-8.3	-8.3
			(25.18)	(25.18)
Prof. Parents: Yes	.	.	1578.587*	1578.587*
			(500.167)	(500.167)
Parent Network: Yes	.	.	1163.197*	1163.197*
			(483.71)	(483.71)
Gender: Male	.	.	467.49	467.49
			(457.239)	(457.239)
N	572	572	525	525
RMSE	5601.282	5601.282	5205.766	5205.766
R^2	0.08	0.08	0.218	0.218
adj R^2	0.076	0.076	0.206	0.206

$*p \leq 0.05$