PS303: Week 10

pp. 472-495

Recap

What does an OLS regression tell us?

Hypothesis tests

0 predictors in null model means our Y estimate is equal to the intercept term (+ any residual left over)

$H_0 : Y_i = b_0 + \epsilon_i$

If our model has K predictors, then we create a regression mode for i data points and K groups.

$H_A : Y_i = (\sum_{k=1}^{K} b_K X_{ik}) + b_0 + \epsilon_i$

To compare H_0 with H_A, we can estimate F-ratios, similar to how we ran the ANOVA. We estimate the sum of squares for the model, which is the difference between the total variance and residual variance.

$$SS_{Model} = SS_{Total} - SS_{Residual}$$

We convert the sums of squares into mean squares by dividing the former with their respective degrees of freedom.

$MS_{Model} = \frac{SS_{Model}}{df_{Model}}$ where $df_{Model} = K$ (number of groups being compared).

$MS_{Residual} = \frac{SS_{Residual}}{df_{Residual}}$ where $df_{Residual} = N - K - 1$ where N and K are the total participants and groups samples respectively.

We can then compute the F-ratio ($F = \frac{MS_{Model}}{MS_{Residual}}$).

> The larger (or smaller) the computed F statistic, the greater (or smaller) is the likelihood of our null hypothesis being *false*.

ID	Physical Activity (minutes)	Depression scores	Age
1	128	48	21
2	127	45	21
3	150	46	22
4	107	41	24
5	142	50	22
6	138	50	20
7	144	41	20
8	133	44	18
9	147	48	25
10	116	50	19
11	90	64	27
12	86	63	27
13	109	48	42
14	105	57	25
15	75	51	37
16	75	58	38
17	100	60	33
18	91	61	43
19	82	62	38
20	79	49	43

```
##
## Call:
## lm(formula = df$depress ~ df$physical + df$age)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.8773  -4.0177   0.5577   4.2361   7.8883
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 75.62516   14.55403   5.196 7.28e-05 ***
## df$physical -0.20009    0.07944  -2.519   0.0221 *
## df$age      -0.05574    0.23429  -0.238   0.8148
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.886 on 17 degrees of freedom
## Multiple R-squared:  0.4289, Adjusted R-squared:  0.3617
## F-statistic: 6.382 on 2 and 17 DF,  p-value: 0.008558
```

Interpreting coefficients

```
##
## Call:
## lm(formula = df$depress ~ df$physical + df$age)
##
## Coefficients:
## (Intercept)  df$physical       df$age
##    75.62516     -0.20009     -0.05574
```

Note that the coefficients for both physical activity ($b_{Physical} = -.20$) and age ($b_{Age} = -.06$) are *negative*. This implies that increased depression scores are associated with reduced physical activity and age.

But are these coefficients reliable, or is the variance due to 'random noise'? We ran one-sample t-tests to estimate whether the null hypothesis that the coefficient is statistically equivalent to 0 can be retained/rejected.

$$H_0 : b_i = 0$$

$$H_A : b_i \neq 0$$

Recall that a linear regression is simply a correlation with a single predictor. We can estimate whether two variables are correlated by applying the `cor.test()` function.

```
cor.test(x=df$depress, y=df$physical)
```

```
##
##  Pearson's product-moment correlation
##
## data:  df$depress and df$physical
## t = -3.6621, df = 18, p-value = 0.001783
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.8501228 -0.2966781
## sample estimates:
##        cor
## -0.6534146
```

We found a significant *negative* correlation between depression scores and physical activity ($r = -.65, p = .002$), as expected. The significant t-statistic for the correlation corresponds with the t-statistic noted for the coefficient across the earlier regression model.

Recall that 95% confidence intervals ($CI_{95\%}$) tell us the min-max range within which a popoulation estimate will be observed. For example, $M = 2, CI_{95\%} : 1.7_{to}2.1$ tells us that the population parameter has a a 95% probability of falling between 1.7 and 2.1. A confidence interval helps us locate the true population parameter (μ) as the point estimate we typically calculate is a *sample* (therefore biased) parameter (M).

Since our regression coefficients (b_i) are also sample parameters, they do not state anything about the true population values by themselves. Hence we can construct confidence intervals for coefficients as follows:

$$CI_b = \hat{b} \pm (t_{Critical} \times SE(\hat{b}))$$

> The $t_{Critical}$ value is the test statistic at the 2.5% and 97.5% limits of the t-distribution with $N - 1 - K$ degrees of freedom when we are running a two-sided test. For a one-sided test, the critical value would be at the 5% *or* 95% limits, depending on the direction we are testing.

We can apply the `confint()` function within `R` to estimate the confidence intervals of our coefficients

```
confint(object = mod2,   # Assign the regression model into 'object'
        level = .95)    # Declare the confidence interval range
```

```
##                  2.5 %       97.5 %
## (Intercept) 44.9188456 106.33147593
## df$physical -0.3677007  -0.03248664
## df$age      -0.5500581   0.43857177
```

If we wanted a 90% interval, we would alter our `level` accordingly

```
confint(object=mod2,level=.9)
```

```
##                   5 %         95 %
## (Intercept) 50.3068756 100.94344591
## df$physical -0.3382907  -0.06189661
## df$age      -0.4633208   0.35183445
```

> Note the ranges for b_{Age} pass through the null estimate (0). This tells us that there is a 95% (or 90%) probability that the population mean is 0, in which case we **retain** the null ($b_i = 0$). This is **not** the case for $b_{Physical}$.

Our two predictors (`physical activity` and `age`) represent different scales relative to each other, and relative to our outcome variable `Depression` . The variability across scales render interpretation of coefficients difficult - larger scales can bias coefficients outputs (e.g., minutes for `physical activity` vs years for `age`).

We can standardize our coefficients ($b \rightarrow \beta$) to estimate which predictors have the strongest relationship with the outcome while controlling for within-scale variance. Standardization constrains the entire σ of our coefficients within 1.

$$\beta_{Predictor} = b_{Predictor} \times \frac{\sigma_{Predictor}}{\sigma_{Outcome}}$$

We already estimated $b_{Physical} = -.20$. We can estimate the standard deviations (σ's) of `physical activity` (*Predictor*) and `depression` (*Outcome*):

`sd(df$depress) = 7.37` which is the $\sigma_{Outcome}$.
`sd(df$physical) = 25.9` which is the $\sigma_{Predictor}$.

We can estimate our standardized coefficient by applying it to the formula above:

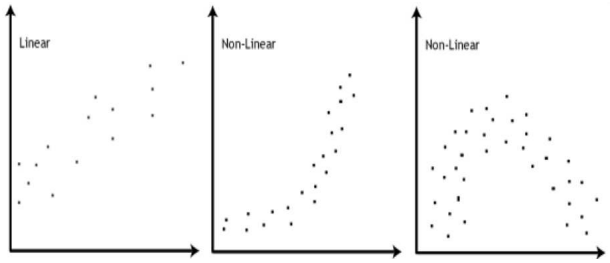
$$-.20 \times \frac{25.9}{7.37} = -.703$$

Our standardized coefficient for `physical activity` is $\beta_{Physical} = -.703$.

Assumptions before running a regression

1. *Normality:* The residuals ($\epsilon_i = \hat{Y}_i - Y_i$) should be normally distributed.
2. *Linearity:* The data should be linearly related.
3. *Homogeneity of variance:* The standard deviation of residuals (σ_ϵ) should be statistically equivalent.
4. *Collinearity:* When you use multiple predictors, these should not strongly correlate with each other.
5. *No extreme outliers:* A couple of data points are not distorting the model.

Examples of linear and non-linear trends



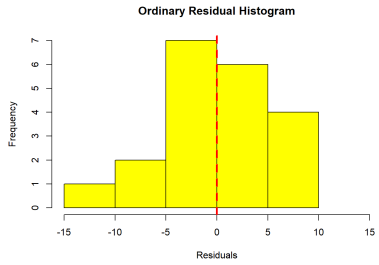
Residuals & Normality

Much of our assumption checks involve estimating patterns across our model residuals (ϵ). We already discussed that 'ordinary' residuals are the difference between the predicted ($\hat{Y}$) and observed (Y) estimates. That is, for each i data point, the ordinary residual is $\epsilon_i = \hat{Y}_i - Y_i$. We can extract the residuals from our model by applying the `residuals()` function.

`residuals(mod2)=`

`-0.84, -4.04, 1.62, -11.88, 4.01, 3.1, -4.7, -4.01, 3.18, -1.36, 7.89, 6.09, -3.47, 3.78, -7.56, -0.5, 6.22, 5.98, 4.9, -8.42`

We can check the distribution of the residuals with a histogram. We can also run a `shapiro test for normality` to acquire a quantitative estimate.




```
shapiro.test(ord.res)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  ord.res
## W = 0.94698, p-value = 0.3235
```

Similar to the case with coefficients, if residuals of predictors and outcomes are from different scales, we can *standardize* them to facilitate interpretation. The formula is

$$\epsilon_i^s = \frac{\epsilon_i}{\sigma\sqrt{1-h_i}}$$

.

A third form ("jackknifed residuals") may be applied if we have many outliers. In this case, they may be preferable to standardized residuals. The formula is:

$$\epsilon_i^* = \frac{\epsilon_i}{\sigma_{-i}\sqrt{\hat{1-h_i}}}$$

$\hat{\sigma_{-i}}$ is the "estimate of the residual standard deviation that *would have been obtained* if the ith data point was removed." This can be estimated by

$$\hat{\sigma_{-i}} = \hat{\sigma}\sqrt{\frac{N-K-1-\epsilon_i^{*2}}{N-K-2}}$$

Fortunately, we can simply apply the `rstandard()` and `rstudent()` functions.

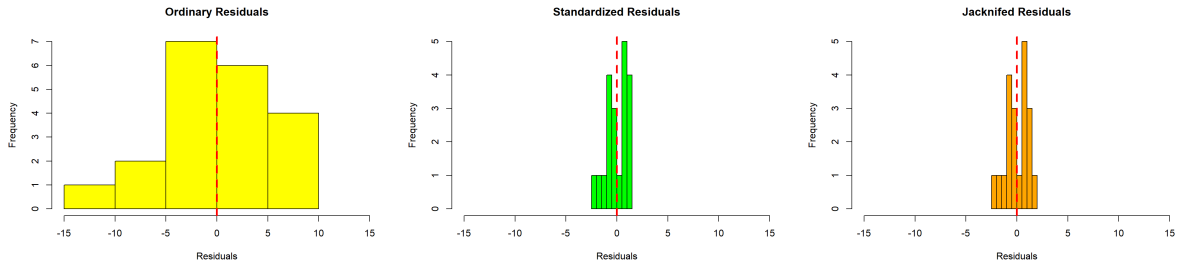
`rstandard(mod2)=`

`-0.15, -0.72, 0.3, -2.12, 0.73, 0.56, -0.86, -0.73, 0.61, -0.25, 1.46, 1.15, -0.72, 0.67, -1.39, -0.09, 1.09, 1.16, 0.89, -1.6`

`rstudent(mod2)=`

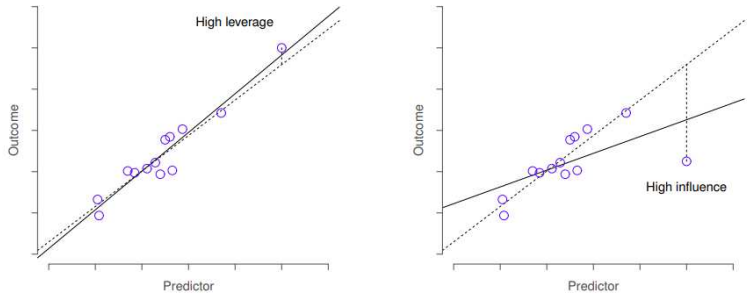
`-0.15, -0.71, 0.3, -2.4, 0.72, 0.55, -0.85, -0.72, 0.6, -0.24, 1.51, 1.16, -0.71, 0.66, -1.44, -0.09, 1.1, 1.17, 0.88, -1.68`

We can look at the histograms from the three varieties of residuals obtained



Outliers

> Any observation that varies notably from the model predictions (usually associated with jackknifed residuals that deviate notably). Outliers which have high **leverage** fall along predicted trends while varying notably from the remaining set of distributions. When an outlier varies notably from observed values *and* predicted trends, they have high **influence**.



Leverage is presented as *hat value* estimates, or h_i. Influence is measured through estimating Cook's distances (D_i) where

$$D_i = \frac{\epsilon_i^{*2}}{K+1} \times \frac{h_i}{1-h_i}$$

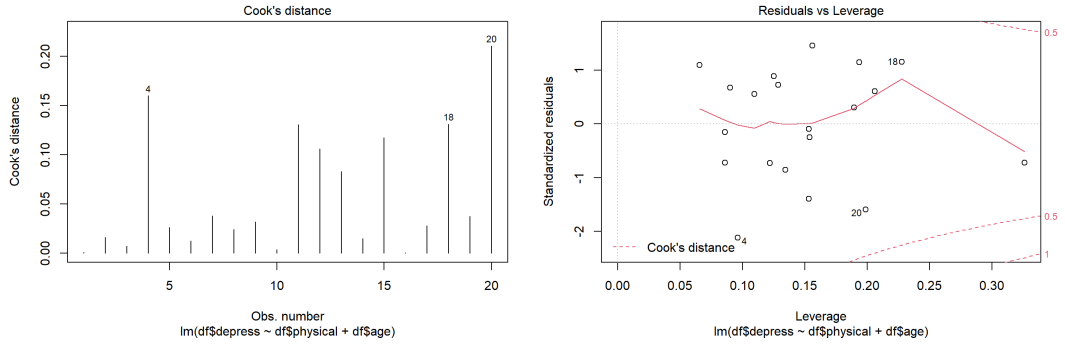
There are functions within R to extract leverage (`hatvalues(model)`) and influence (`cooks.distance(model)`) metrics.

```
hatvalues(mod2)=
0.0859411, 0.0858809, 0.1895072, 0.0963048, 0.1286199, 0.1094105, 0.1344169, 0.1218646, 0.2058641, 0.1537667, 0.1558316, 0.1936944, 0.3259137,
```

```
cooks.distance(mod2)=
7.0257884\times 10^{-4}, 0.0161607, 0.0072417, 0.1600575, 0.0262658, 0.0127761, 0.0380784, 0.0244418, 0.0318038, 0.0037941, 0.1309195, 0.10624
```

We can estimate whether any of the outliers have significant leverage (h_i) and/or influence (D_i) by passing values to the `which=` argument when plotting our regression.

```
plot(x=mod2,which=4)  # Cook's distance # Estimating Influence
plot(x=mod2,which=5)  # Hat values # Estimating Leverage
```



There are at least **three** data points that have high leverage and influence. We can re-run our regression model by excluding said data to note any changes in the model.

```
mod3 <-  lm(df$depress~df$physical+df$age,
          subset= c(-4,-18,-20))# Identify outlier positions
mod3
```

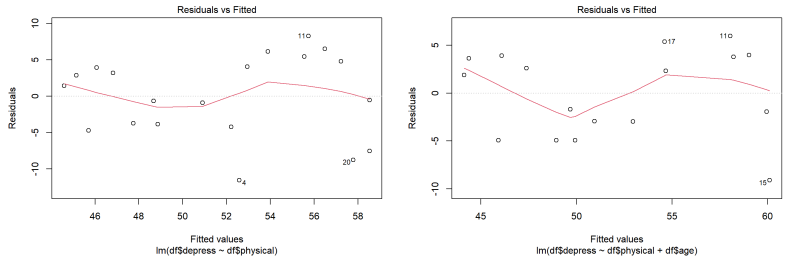
```
##
## Call:
## lm(formula = df$depress ~ df$physical + df$age, subset = c(-4,
##     -18, -20))
##
## Coefficients:
## (Intercept)  df$physical       df$age
##     84.5185      -0.2456      -0.1618
```

This does not vary noticeably from our original model (without outliers removed)

```
##
## Call:
## lm(formula = df$depress ~ df$physical + df$age)
##
## Coefficients:
## (Intercept)  df$physical       df$age
##    75.62516     -0.20009     -0.05574
```

For the data to be linear, the predicted/fitted values ($\hat{Y}_i$) should covary with residuals (ϵ_i) across a reasonably straight line.

```
plot(x=mod1,which=1)  # Fitted & residual values with outliers present
plot(x=mod3,which=1)  # Fitted & residual values with outliers absent
```

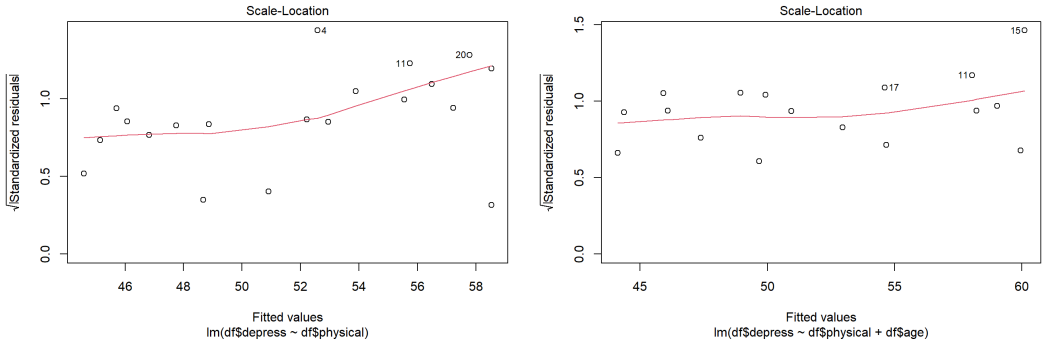


The relationship appears reasonably linear regardless of whether outliers are present or not.

Homogeneity of variance

Check whether variances are linear (constant) across standardized residuals relative to fitted values

```
# Are variances homogenous?
plot(x=mod1,which=3)
plot(x=mod3,which=3)
```



Variances do *not* appear homogeneous, hence our t-tests (across b_i coefficients) may not have been reliable.

We can reduce hetereogenity across variances by applying a *hccm* (**heteroscedasticity corrected covariance matrix**) when estimating standard error.

```
require(lmtest)                          # Package
lmod1 <- lmtest::coeftest(mod1,type=hccm)  # Test coefficients
lmod1
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 72.464437   5.786419 12.5232 2.531e-10 ***
## df$physical -0.185831   0.050745 -3.6621  0.001783 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The coefficient remains significantly different from the null after the *hccm* is applied

Check whether predictors are associated with each other (relevant whenever predictors > 2). To do this, we estimate **variance inflation factors (VIFs)** which provide a quantitative indicator of predictor collinearity. To estimate **VIF** for each predictor X_k in our model, we can run the following:

$$VIF_k = \frac{1}{1 - R^2_{(-k)}}$$

$R^2_{(-k)}$ is the R^2 estimate if we were to run a regression where X_k is the outcome variable and all X_{-k} were the predictors. **VIF**'s smaller than 5 are typically fine (ie the predictors are not **too** strongly correlated).

```
car::vif(mod = mod2)
```

```
## df$physical      df$age
##    2.322395    2.322395
```

> -*"…the square root (of the VIF) tells us how wide the confidence interval for the corresponding coefficient is, relative to what would be expected if the predictors were uncorrelated with one another. With two predictors, the VIF values are always going to be the same…"* - p. 489

Selecting between models

Recall that we estimated two models

```
##
## Call:
## lm(formula = df$depress ~ df$physical)
##
## Coefficients:
## (Intercept)  df$physical
##     72.4644      -0.1858
```

```
##
## Call:
## lm(formula = df$depress ~ df$physical + df$age)
##
## Coefficients:
## (Intercept)  df$physical       df$age
##    75.62516     -0.20009     -0.05574
```

We want to select the "best" model (include variables that function as significant predictors). Including multiple predictors will boost your R^2 but at the cost of reduced generalizability to new observations.

For linear models, we can estimate the **AIC** (*Akaike Information Criteria*) for each model. The smaller the **AIC**, the better the model.

Using *backward elimination*, we start with the full model and incrementally 'drop' predictors. The *model* with the smallest **AIC** is the one we want. Alternatively, we can specify the largest model we are willing to tolerate and use *forward selection*.

```
step(object=mod2,direction="backward")
```

```
## Start:  AIC=73.65
## df$depress ~ df$physical + df$age
##
##               Df Sum of Sq    RSS    AIC
## - df$age       1     1.961 590.93 71.719
## <none>                     588.97 73.653
## - df$physical  1   219.793 808.76 77.995
##
## Step:  AIC=71.72
## df$depress ~ df$physical
##
##               Df Sum of Sq     RSS    AIC
## <none>                      590.93 71.719
## - df$physical  1    440.27 1031.20 80.855
```

```
##
## Call:
## lm(formula = df$depress ~ df$physical)
##
## Coefficients:
## (Intercept)  df$physical
##     72.4644      -0.1858
```

```
null.model <- lm(depress ~ 1, df) # Null model with intercept only

# Forward selection using base R
step(object = null.model,
     direction = "forward",
     scope = depress ~ physical + age)
```

```
## Start:  AIC=80.85
## depress ~ 1
##
##            Df Sum of Sq     RSS    AIC
## + physical  1    440.27  590.93 71.719
## + age       1    222.44  808.76 77.995
## <none>                  1031.20 80.855
##
## Step:  AIC=71.72
## depress ~ physical
##
##        Df Sum of Sq    RSS    AIC
## <none>              590.93 71.719
## + age   1    1.9611 588.97 73.653
```

```
##
## Call:
## lm(formula = depress ~ physical, data = df)
##
## Coefficients:
## (Intercept)     physical
##     72.4644      -0.1858
```

Statistical tests between models

We can subject entire models to an analysis of variance (ANOVA) to note whether the difference between the two vary significantly from a null model.

```
anova(mod1,mod2)  # Run ANOVA across the models
```

```
## Analysis of Variance Table
##
## Model 1: df$depress ~ df$physical
## Model 2: df$depress ~ df$physical + df$age
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     18 590.93
## 2     17 588.97  1    1.9611 0.0566 0.8148
```

The models are **not** statistically different, indicating the inclusion of the *Age* predictor did not improve our model, hence we can retain the *Physical Activity* predictor only.

Conclusion

There are numerous diagnostics available for regression models. You don't need to apply them exhaustively, but it's a good idea to have a grasp of what each diagnostic parameter tells us.

Regressions are an excellent "first step" towards identifying meaningful predictors across your data set. By comparing between models and the number of predictors, we can identify the most efficient model (ie that can generalizehas the largest probability of estimating variances across novel observations).

For this week's lab, you will be running a regression model with multiple predictors. Your task is to select the best model (optimal combination of predictors) and provide regression diagnostics. Additional information is provided in the following slides.

In the Physical Activity Scale, there are three variants of Physical Acitivity (vigorous, moderate, walking). Simulated data for these three physical measures, along with depression scores, are provided below. You will assess which combination of predictors is the *best* model for predicting depression scores. For your chosen model, run regression diagnostics to respectively assess whether the assumptions of residual normality, collinearity and homogeneity of variances are met.

1. Assign the three physical activity variables, depression scores and id's to a single data frame (*Hint*: Use the `cbind.data.frame()` function to combine the five variables).
2. Run an OLS regression using the `lm()` function to model `Outcome~ Predictor1+Predictor2+Predictor3` . Next, report whether the model is significant (*Hint*: Use the `summary()` function on the linear model created above).
3. Using backward elimination, report which combination of predictors produces the smallest AIC (*Hint*: Use the `step()` function)
4. Generate diagnostic plots for estimating whether the assumptions for collinearity and homogeneity of variances are met (*Hint*: use the `which=` argument within the `plot()` function).
5. Generate a histogram of ordinary residuals (*Hint*: Use the `residuals()` function on the model created)

Provide all code and plots on a document file (.doc or .pdf) then submit by the end of the following week.

Variables to create

```
# Variable setup
ID         <- seq(1:20)
Walking    <- c(245,270,209,328,108,269,380,206,120,402,200,93,259,136,85,183,291,261,244,305)
Moderate   <- c(74,113,118,51,121,126,132,58,109,54,62,151,59,99,103,146,142,65,100,70)
Vigorous   <- c(54,78,62,76,67,63,35,39,74,55,31,30,61,51,79,69,60,57,75,32)
Depression <- c(44,64,60,57,54,62,53,48,49,58,45,56,43,63,50,34,32,47,40,69)
```

ID	Depression scores	Walking*	Moderate*	Vigorous*
1	44	245	74	54
2	64	270	113	78
3	60	209	118	62
4	57	328	51	76
5	54	108	121	67
6	62	269	126	63
7	53	380	132	35
8	48	206	58	39
9	49	120	109	74
10	58	402	54	55
11	45	200	62	31
12	56	93	151	30
13	43	259	59	61
14	63	136	99	51
15	50	85	103	79
16	34	183	146	69
17	32	291	142	60
18	47	261	65	57
19	40	244	100	75
20	69	305	70	32

Note: *All activity scores in minutes