Using ADNI data for Cusp model fitting

Or you can use this alternative method

Simple way

Auditory Verbal Learning Test fitted with Whole gray matter and covariables.

```
 # Estevez-Gonzalez, A., Kulisevsky, J., Boltes, A., Otermin, P., & Garcia-Sanchez, C. (2003).
 # Rey verbal learning test is a useful tool for differential diagnosis in the preclinical phase
 # of Alzheimer's disease: comparison with mild cognitive impairment and normal aging.
 # International Journal of Geriatric Psychiatry. 18 (11), 1021.
```

```
library("ADNIMERGE")
library(cusp)
library(psych) #for composite scores
# Let's get the data
tmp_np <- merge(adas, neurobat, by=c("RID", "VISCODE") )
mt2fa <- merge(tmp_np, adnimerge, by=c("RID", "VISCODE") )
rm(tmp_np)
# Calculate the subject age at every point
mt2fa$vAGE = mt2fa$AGE + mt2fa$Years
data <- data.frame(mt2fa$WholeBrain, mt2fa$ICV, mt2fa$vAGE, mt2fa$PTGENDER, mt2fa$PTEDUCAT, mt2fa$AVDEL30MIN, mt2fa$AVDELTOT)
datac <- data[complete.cases(data),]
datac$WB = datac$mt2fa.WholeBrain/datac$mt2fa.ICV
fit_avd <- cusp(y ~ mt2fa.AVDEL30MIN, alpha ~ WB +mt2fa.vAGE + mt2fa.PTGENDER +mt2fa.PTEDUCAT, beta ~ WB +mt2fa.vAGE + mt2fa.PTGENDER +mt2fa.PTEDUCAT, datac)
summary(fit_avd)
```

Amazing results

```
Call:
cusp(formula = y ~ mt2fa.AVDEL30MIN, alpha = alpha ~ WB + mt2fa.vAGE +
    mt2fa.PTGENDER + mt2fa.PTEDUCAT, beta = beta ~ WB + mt2fa.vAGE +
    mt2fa.PTGENDER + mt2fa.PTEDUCAT, data = datac)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.06955  -0.26210  -0.03226   0.63723   3.40775

Coefficients:
```

```
                         Estimate Std. Error  z value Pr(>|z|)
a[(Intercept)]          -5.842305   0.325414  -17.953  < 2e-16 ***
a[WB]                    5.365145   0.296437   18.099  < 2e-16 ***
a[mt2fa.vAGE]            0.004777   0.002000    2.388   0.0169 *
a[mt2fa.PTGENDERFemale]  0.364218   0.026875   13.552  < 2e-16 ***
a[mt2fa.PTEDUCAT]        0.078235   0.005217   14.995  < 2e-16 ***
b[(Intercept)]           7.001814   0.509934   13.731  < 2e-16 ***
b[WB]                   -5.970685   0.470236  -12.697  < 2e-16 ***
b[mt2fa.vAGE]           -0.026695   0.003309   -8.069 7.11e-16 ***
b[mt2fa.PTGENDERFemale]  0.395185   0.044766    8.828  < 2e-16 ***
b[mt2fa.PTEDUCAT]        0.033672   0.008002    4.208 2.58e-05 ***
w[(Intercept)]          -1.763659   0.012301 -143.381  < 2e-16 ***
w[mt2fa.AVDEL30MIN]      0.257004   0.001838  139.800  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


  Null deviance:   8677.7  on 6754  degrees of freedom
Linear deviance: 110910.4  on 6749  degrees of freedom
Logist deviance:       NA  on   NA  degrees of freedom
 Delay deviance:   3610.5  on 6743  degrees of freedom

             R.Squared     logLik npar      AIC     AICc      BIC
Linear model 0.1557933 -19036.661    6 38085.32 38085.33 38126.23
Cusp model   0.6142339  -7321.477   12 14666.95 14667.00 14748.77
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 2.343e+04, df = 6, p-value = 0

Number of optimization iterations: 40
```

Z-scores

Now let's compare the weights of each variable on the model. We need to translate everything to z-scores (or just do another linear transformation that carry every thing to comparable values)

```
datac$zWB = (datac$WB - mean(datac$WB))/sd(datac$WB)
datac$zAge = (datac$mt2fa.vAGE -
mean(datac$mt2fa.vAGE))/sd(datac$mt2fa.vAGE)
datac$zEduc = (datac$mt2fa.PTEDUCAT -
mean(datac$mt2fa.PTEDUCAT))/sd(datac$mt2fa.PTEDUCAT)
datac$zAVD = (datac$mt2fa.AVDEL30MIN -
mean(datac$mt2fa.AVDEL30MIN))/sd(datac$mt2fa.AVDEL30MIN)
fit_avd_z <- cusp(y ~ zAVD, alpha ~ zWB + zAge + mt2fa.PTGENDER + zEduc,
beta ~ zWB +zAge + mt2fa.PTGENDER + zEduc, datac)
```

```
summary(fit_avd_z)
```

The results are of course the same but the coefficients must be meaningful now,

```
Coefficients:
                            Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]             -0.708321   0.023240 -30.478  < 2e-16 ***
a[zWB]                      0.290456   0.016048  18.099  < 2e-16 ***
a[zAge]                     0.034254   0.014342   2.388   0.0169 *
a[mt2fa.PTGENDERFemale]     0.364218   0.026875  13.552  < 2e-16 ***
a[zEduc]                    0.223024   0.014873  14.995  < 2e-16 ***
b[(Intercept)]              1.603759   0.046739  34.313  < 2e-16 ***
b[zWB]                     -0.323238   0.025457 -12.697  < 2e-16 ***
b[zAge]                    -0.191434   0.023726  -8.069 7.11e-16 ***
b[mt2fa.PTGENDERFemale]     0.395185   0.044766   8.828  < 2e-16 ***
b[zEduc]                    0.095988   0.022812   4.208 2.58e-05 ***
w[(Intercept)]             -0.688350   0.009603 -71.682  < 2e-16 ***
w[zAVD]                     1.133500   0.008108 139.800  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Composite scores

First I'm going to try another NP test (Recognition)

```
fit_avr <- cusp(y ~ zAVR, alpha ~ zWB + zAge + mt2fa.PTGENDER + zEduc, beta
~ zWB +zAge + mt2fa.PTGENDER + zEduc, datac)
```

and this is not so good but still an improvement is done

```
> summary(fit_avr)
Call:
cusp(formula = y ~ zAVR, alpha = alpha ~ zWB + zAge + mt2fa.PTGENDER +
    zEduc, beta = beta ~ zWB + zAge + mt2fa.PTGENDER + zEduc,
    data = datac)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.3337  -0.8146  -0.1929   0.2446   2.5763

Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]              0.74119    0.02784  26.625  < 2e-16 ***
a[zWB]                      0.40520    0.01943  20.854  < 2e-16 ***
a[zAge]                     0.08352    0.01681   4.967 6.79e-07 ***
a[mt2fa.PTGENDERFemale]    -0.09332    0.03128  -2.983  0.00285 **
a[zEduc]                    0.10035    0.01495   6.713 1.91e-11 ***
```

```
b[(Intercept)]           1.09138    0.05476  19.932  < 2e-16 ***
b[zWB]                   0.02940    0.02921   1.007  0.31416
b[zAge]                  0.11858    0.02729   4.345 1.39e-05 ***
b[mt2fa.PTGENDERFemale]  0.53540    0.04991  10.726  < 2e-16 ***
b[zEduc]                 0.11832    0.02474   4.782 1.74e-06 ***
w[(Intercept)]           0.71871    0.01153  62.322  < 2e-16 ***
w[zAVR]                  0.99105    0.00889 111.482  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


  Null deviance: 6633.7  on 6754  degrees of freedom
Linear deviance: 5952.1  on 6749  degrees of freedom
Logist deviance:     NA  on   NA  degrees of freedom
 Delay deviance: 5049.6  on 6743  degrees of freedom

             R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.1187225 -9157.572    6 18327.14 18327.16 18368.05
Cusp model   0.3758839 -7966.518   12 15957.04 15957.08 16038.85
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 2382, df = 6, p-value = 0

Number of optimization iterations: 38
```

Now, let's try a composite score

```
gfam <- data.frame(datac$zAVD, datac$zAVR)
famod <- fa(gfam, scores="regression")
datac$cs <- famod$scores
fit_cs <- cusp(y ~ cs, alpha ~ zWB + zAge + mt2fa.PTGENDER + zEduc, beta ~
zWB +zAge + mt2fa.PTGENDER + zEduc, datac)
```

And we get a very bad fit result

```
> summary(fit_cs)

Call:
cusp(formula = y ~ cs, alpha = alpha ~ zWB + zAge + mt2fa.PTGENDER +
    zEduc, beta = beta ~ zWB + zAge + mt2fa.PTGENDER + zEduc,
    data = datac)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9864  -0.5145   0.0386   0.5796   2.8034
```

```
Coefficients:
                         Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]          -0.166529   0.015351 -10.848  < 2e-16 ***
a[zWB]                   0.508523   0.009612  52.905  < 2e-16 ***
a[zAge]                  0.124850   0.017119   7.293 3.03e-13 ***
a[mt2fa.PTGENDERFemale]  0.251292   0.020108  12.497  < 2e-16 ***
a[zEduc]                 0.230918         NA      NA       NA
b[(Intercept)]          -0.224522   0.022933  -9.791  < 2e-16 ***
b[zWB]                  -0.231656   0.010449 -22.171  < 2e-16 ***
b[zAge]                 -0.128716   0.012640 -10.183  < 2e-16 ***
b[mt2fa.PTGENDERFemale]  0.845789   0.017774  47.587  < 2e-16 ***
b[zEduc]                 0.204127   0.007479  27.293  < 2e-16 ***
w[(Intercept)]          -0.035730   0.011210  -3.187  0.00144 **
w[cs]                    1.012113   0.006090 166.187  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


  Null deviance: 5487.3  on 6754  degrees of freedom
Linear deviance: 4490.5  on 6749  degrees of freedom
Logist deviance:     NA  on   NA  degrees of freedom
 Delay deviance: 5324.3  on 6743  degrees of freedom

              R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.16171127 -8205.808    6 16423.62 16423.63 16464.52
Cusp model   0.03112304 -8567.331   12 17158.66 17158.71 17240.48
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 723, df = 6, p-value = 0

Number of optimization iterations: 106
```

That is, the composite score is not related through a cusp model to the independent variable analyzed here

A try for ADAS-Cog

```
data <- data.frame(mt2fa$WholeBrain, mt2fa$ICV, mt2fa$vAGE, mt2fa$PTGENDER,
mt2fa$PTEDUCAT, mt2fa$Q4SCORE, mt2fa$Q8SCORE)
datac <- data[complete.cases(data),]
datac$WB = datac$mt2fa.WholeBrain/datac$mt2fa.ICV
datac$zWB = (datac$WB - mean(datac$WB))/sd(datac$WB)
datac$zAge = (datac$mt2fa.vAGE -
mean(datac$mt2fa.vAGE))/sd(datac$mt2fa.vAGE)
```

```
datac$zEduc = (datac$mt2fa.PTEDUCAT -
mean(datac$mt2fa.PTEDUCAT))/sd(datac$mt2fa.PTEDUCAT)
datac$dr = (mean(datac$mt2fa.Q4SCORE) -
datac$mt2fa.Q4SCORE)/sd(datac$mt2fa.Q4SCORE)
datac$r = (mean(datac$mt2fa.Q8SCORE) -
datac$mt2fa.Q8SCORE)/sd(datac$mt2fa.Q8SCORE)
fit_dr <- cusp(y ~ dr, alpha ~ zWB + zAge + mt2fa.PTGENDER + zEduc, beta ~
zWB +zAge + mt2fa.PTGENDER + zEduc, datac)
fit_r <- cusp(y ~ r, alpha ~ zWB + zAge + mt2fa.PTGENDER + zEduc, beta ~ zWB
+zAge + mt2fa.PTGENDER + zEduc, datac)
```

not bad at all for Delay Recall

```
> summary(fit_dr)

Call:
cusp(formula = y ~ dr, alpha = alpha ~ zWB + zAge + mt2fa.PTGENDER +
    zEduc, beta = beta ~ zWB + zAge + mt2fa.PTGENDER + zEduc,
    data = datac)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1117  -0.4991  -0.1005   0.4315   3.3147

Coefficients:
                         Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]           0.007960   0.020930   0.380 0.703714
a[zWB]                   0.508635   0.017192  29.585  < 2e-16 ***
a[zAge]                  0.138990   0.013981   9.942  < 2e-16 ***
a[mt2fa.PTGENDERFemale]  0.102236   0.025494   4.010 6.07e-05 ***
a[zEduc]                 0.190303   0.013197  14.420  < 2e-16 ***
b[(Intercept)]           0.825442   0.054864  15.045  < 2e-16 ***
b[zWB]                  -0.235399   0.029643  -7.941 2.01e-15 ***
b[zAge]                 -0.182519   0.025867  -7.056 1.71e-12 ***
b[mt2fa.PTGENDERFemale]  0.745478   0.047279  15.768  < 2e-16 ***
b[zEduc]                 0.125539   0.023628   5.313 1.08e-07 ***
w[(Intercept)]           0.039500   0.011097   3.559 0.000372 ***
w[dr]                    1.118457   0.009343 119.717  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1



  Null deviance: 8511.4  on 6804  degrees of freedom
Linear deviance: 5443.0  on 6799  degrees of freedom
Logist deviance:     NA  on   NA  degrees of freedom
 Delay deviance: 4560.9  on 6793  degrees of freedom

             R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.2000289 -8896.008    6 17804.02 17804.03 17844.97
Cusp model   0.4672353 -7913.728   12 15851.46 15851.50 15933.36
```

```
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 1965, df = 6, p-value = 0

Number of optimization iterations: 38
```

but worst for Recognition

```
> summary(r)

Call:
cusp(formula = y ~ r, alpha = alpha ~ zWB + zAge + mt2fa.PTGENDER +
    zEduc, beta = beta ~ zWB + zAge + mt2fa.PTGENDER + zEduc,
    data = datac)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1343  -0.8453  -0.2669   0.2033   2.5628

Coefficients:
                         Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]           0.754786   0.029939  25.211  < 2e-16 ***
a[zWB]                   0.445183   0.020237  21.998  < 2e-16 ***
a[zAge]                  0.082755   0.017131   4.831 1.36e-06 ***
a[mt2fa.PTGENDERFemale] -0.138675   0.032093  -4.321 1.55e-05 ***
a[zEduc]                 0.076295   0.015311   4.983 6.26e-07 ***
b[(Intercept)]           0.898577   0.061474  14.617  < 2e-16 ***
b[zWB]                   0.013364   0.030828   0.433 0.664652
b[zAge]                  0.041912   0.028325   1.480 0.138959
b[mt2fa.PTGENDERFemale]  0.395628   0.052501   7.536 4.86e-14 ***
b[zEduc]                 0.084819   0.025534   3.322 0.000894 ***
w[(Intercept)]           0.641352   0.012443  51.543  < 2e-16 ***
w[r]                     0.960657   0.009485 101.286  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


  Null deviance: 6279.2  on 6804  degrees of freedom
Linear deviance: 5963.7  on 6799  degrees of freedom
Logist deviance:     NA  on    NA  degrees of freedom
 Delay deviance: 5578.1  on 6793  degrees of freedom

             R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.1235032 -9206.852    6 18425.70 18425.72 18466.66
Cusp model   0.2913304 -8250.294   12 16524.59 16524.63 16606.49
---
```

```
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 1913, df = 6, p-value = 0

Number of optimization iterations: 44
```

Notas para Composite Scores

Lo ideal seria hacer script con todos los composites posibles y mirarlo contra los biomarcadores disponibles en *adnimerge*. Pero cada biomarcador lleva un tipo de procesamiento *distinto* y cada composite ha de ser definido previamente. Por ejemplo el composite de Delay Recall (drcs) lo construimos a partir de **adas.Q4SCORE** y **neurobat.AVDEL30MIN** pero en cada caso hay que definir las variables de partida.

Hay varios biomarcadores en la tabla adnimerge que pueden estar relacionados con los composites neuropsicologicos

```
> names(adnimerge)
 [1] "ORIGPROT"                "COLPROT"                 "RID"
"PTID"
 [5] "VISCODE"                 "EXAMDATE"                "SITE"
"DX.bl"
 [9] "AGE"                     "PTGENDER"                "PTEDUCAT"
"PTETHCAT"
[13] "PTRACCAT"                "PTMARRY"                 "APOE4"
"FDG"
[17] "PIB"                     "AV45"                    "CDRSB"
"ADAS11"
[21] "ADAS13"                  "MMSE"                    "RAVLT.immediate"
"RAVLT.learning"
[25] "RAVLT.forgetting"        "RAVLT.perc.forgetting"   "FAQ"
"MOCA"
[29] "EcogPtMem"               "EcogPtLang"              "EcogPtVisspat"
"EcogPtPlan"
[33] "EcogPtOrgan"             "EcogPtDivatt"            "EcogPtTotal"
"EcogSPMem"
[37] "EcogSPLang"              "EcogSPVisspat"           "EcogSPPlan"
"EcogSPOrgan"
[41] "EcogSPDivatt"            "EcogSPTotal"             "FLDSTRENG"
"FSVERSION"
[45] "Ventricles"              "Hippocampus"             "WholeBrain"
"Entorhinal"
[49] "Fusiform"                "MidTemp"                 "ICV"
"DX"
[53] "EXAMDATE.bl"             "CDRSB.bl"                "ADAS11.bl"
"ADAS13.bl"
```

```
[57] "MMSE.bl"                    "RAVLT.immediate.bl"
"RAVLT.learning.bl"        "RAVLT.forgetting.bl"
[61] "RAVLT.perc.forgetting.bl" "FAQ.bl"                    "FLDSTRENG.bl"
"FSVERSION.bl"
[65] "Ventricles.bl"            "Hippocampus.bl"            "WholeBrain.bl"
"Entorhinal.bl"
[69] "Fusiform.bl"              "MidTemp.bl"                "ICV.bl"
"MOCA.bl"
[73] "EcogPtMem.bl"             "EcogPtLang.bl"
"EcogPtVisspat.bl"         "EcogPtPlan.bl"
[77] "EcogPtOrgan.bl"           "EcogPtDivatt.bl"           "EcogPtTotal.bl"
"EcogSPMem.bl"
[81] "EcogSPLang.bl"            "EcogSPVisspat.bl"          "EcogSPPlan.bl"
"EcogSPOrgan.bl"
[85] "EcogSPDivatt.bl"          "EcogSPTotal.bl"            "FDG.bl"
"PIB.bl"
[89] "AV45.bl"                  "Years.bl"                  "Month.bl"
"Month"
[93] "M"
```

El problema es que cada uno debe ser analizado de manera distinta. Las variables *Ventricles*, *Hippocampus* y *WholeBrain* deben de alguna manera normalizarse por *ICV* (revisar *Entorhinal*, *Fusiform* y *MidTemp*) mientras que *FDG*, *PIB* y *AV45* son variables normalizadas.

```
library("ADNIMERGE")
library(cusp)
library(psych) #for composite scores
# Let's get the data
tmp_np <- merge(adas, neurobat, by=c("RID", "VISCODE") )
m <- merge(tmp_np, adnimerge, by=c("RID", "VISCODE") )
rm(tmp_np)
# Select data
m$cAGE = m$AGE + m$Years
data <- data.frame(m$WholeBrain, m$ICV, m$cAGE, m$PTGENDER, m$PTEDUCAT,
m$AVDEL30MIN, m$Q4SCORE)
datac <- data[complete.cases(data),]
#Z-scores and Composite Scores
datac$zavd = (datac$m.AVDEL30MIN -
mean(datac$m.AVDEL30MIN))/sd(datac$m.AVDEL30MIN)
datac$zdr = (mean(datac$m.Q4SCORE) - datac$m.Q4SCORE)/sd(datac$m.Q4SCORE)
datac$zAge = (datac$m.cAGE - mean(datac$m.cAGE))/sd(datac$m.cAGE)
datac$zEduc = (datac$m.PTEDUCAT -
mean(datac$m.PTEDUCAT))/sd(datac$m.PTEDUCAT)
gfam <- data.frame(datac$zavd, datac$zdr)
famod <- fa(gfam, scores="regression")
datac$drcs <- famod$scores
# NI biomarker
datac$wb = datac$m.WholeBrain/datac$m.ICV
datac$zwb = (datac$wb - mean(datac$wb))/sd(datac$wb)
#fit to Cusp model
```

```
fit <- cusp(y ~ drcs, alpha ~ zwb + zAge + m.PTGENDER + zEduc, beta ~ zwb
+zAge + m.PTGENDER + zEduc, datac)
summary(fit)
```

El resultado no es demasiado bueno para la materia gris

```
> summary(fit)

Call:
cusp(formula = y ~ drcs, alpha = alpha ~ zwb + zAge + m.PTGENDER +
    zEduc, beta = beta ~ zwb + zAge + m.PTGENDER + zEduc, data = datac)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-3.03128  -0.34504   0.04153   0.68234   3.46997

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]      -0.495490   0.034268 -14.459  < 2e-16 ***
a[zwb]               0.439670   0.016181  27.172  < 2e-16 ***
a[zAge]              0.084987   0.015421   5.511 3.57e-08 ***
a[m.PTGENDERFemale]  0.335385   0.036312   9.236  < 2e-16 ***
a[zEduc]             0.246808   0.016847  14.650  < 2e-16 ***
b[(Intercept)]       0.707160   0.038400  18.416  < 2e-16 ***
b[zwb]              -0.430749   0.019983 -21.556  < 2e-16 ***
b[zAge]             -0.263518   0.015366 -17.149  < 2e-16 ***
b[m.PTGENDERFemale]  0.737773   0.041330  17.851  < 2e-16 ***
b[zEduc]             0.121714   0.021839   5.573 2.50e-08 ***
w[(Intercept)]      -0.343946   0.012126 -28.364  < 2e-16 ***
w[drcs]              1.144778   0.009349 122.443  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


  Null deviance: 7603.1  on 6754  degrees of freedom
Linear deviance: 4650.8  on 6749  degrees of freedom
Logist deviance:     NA  on   NA  degrees of freedom
 Delay deviance: 5097.1  on 6743  degrees of freedom

             R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.1983602 -8324.282    6 16660.56 16660.58 16701.47
Cusp model   0.3583674 -7877.500   12 15779.00 15779.05 15860.82
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 893.6, df = 6, p-value = 0
```

```
Number of optimization iterations: 52
```

Un poco mejor (no mucho) para los Ventriculos

```
> summary(fit)

Call:
cusp(formula = y ~ drcs, alpha = alpha ~ zwb + zAge + m.PTGENDER +
    zEduc, beta = beta ~ zwb + zAge + m.PTGENDER + zEduc, data = datac)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.96597  -0.34590   0.08371   0.71785   3.30025

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]      -0.45714    0.02514 -18.184  < 2e-16 ***
a[zwb]              -0.37215    0.01895 -19.643  < 2e-16 ***
a[zAge]              0.02568    0.01398   1.837   0.0662 .
a[m.PTGENDERFemale]  0.26724    0.02818   9.484  < 2e-16 ***
a[zEduc]             0.24629    0.01497  16.455  < 2e-16 ***
b[(Intercept)]       0.68135    0.06771  10.063  < 2e-16 ***
b[zwb]               0.04897    0.02989   1.638   0.1013
b[zAge]             -0.13237    0.02618  -5.057 4.27e-07 ***
b[m.PTGENDERFemale]  0.71151    0.05476  12.994  < 2e-16 ***
b[zEduc]             0.13563    0.02624   5.169 2.35e-07 ***
w[(Intercept)]      -0.28049    0.01186 -23.655  < 2e-16 ***
w[drcs]              1.12939    0.01124 100.486  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


  Null deviance: 7186.9  on 6554  degrees of freedom
Linear deviance: 4853.0  on 6549  degrees of freedom
Logist deviance:     NA  on   NA  degrees of freedom
 Delay deviance: 4458.2  on 6543  degrees of freedom

             R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.1386990 -8315.823    6 16643.65 16643.66 16684.37
Cusp model   0.4315817 -7959.414   12 15942.83 15942.88 16024.28
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 712.8, df = 6, p-value = 0
```

```
Number of optimization iterations: 40
```

Para el hipocampo el lineal es tan bueno como el no lineal

```
> summary(fit)

Call:
cusp(formula = y ~ drcs, alpha = alpha ~ zwb + zAge + m.PTGENDER +
    zEduc, beta = beta ~ zwb + zAge + m.PTGENDER + zEduc, data = datac)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.0590  -0.3649   0.0143   0.5757   3.2717

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]      -0.5748835  0.0203013 -28.318  < 2e-16 ***
a[zwb]               0.7766157  0.0209173  37.128  < 2e-16 ***
a[zAge]              0.1420567         NA      NA       NA
a[m.PTGENDERFemale]  0.3207513         NA      NA       NA
a[zEduc]             0.2969783  0.0005627 527.749  < 2e-16 ***
b[(Intercept)]       0.5155457         NA      NA       NA
b[zwb]              -0.7661551  0.0139140 -55.064  < 2e-16 ***
b[zAge]             -0.3060879  0.0196256 -15.596  < 2e-16 ***
b[m.PTGENDERFemale]  0.7437734  0.0132296  56.220  < 2e-16 ***
b[zEduc]             0.1028033  0.0284092   3.619 0.000296 ***
w[(Intercept)]      -0.4249690  0.0080047 -53.090  < 2e-16 ***
w[drcs]              1.1575905  0.0062789 184.362  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


  Null deviance: 6826.7  on 5939  degrees of freedom
Linear deviance: 3140.1  on 5934  degrees of freedom
Logist deviance:     NA  on   NA  degrees of freedom
 Delay deviance: 4303.2  on 5928  degrees of freedom

             R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.3836286 -6535.252    6 13082.50 13082.52 13122.64
Cusp model   0.3817421 -6092.774   12 12209.55 12209.60 12289.82
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 885, df = 6, p-value = 0
```

```
Number of optimization iterations: 65
```

Malo para el FDG

```
> summary(fit)

Call:
cusp(formula = y ~ drcs, alpha = alpha ~ zwb + zAge + m.PTGENDER +
    zEduc, beta = beta ~ zwb + zAge + m.PTGENDER + zEduc, data = datac)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-3.02185  -0.35020   0.05687   0.63803   3.60924

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]      -0.508309   0.023910 -21.259  < 2e-16 ***
a[zwb]               0.524366   0.014518  36.119  < 2e-16 ***
a[zAge]             -0.077225   0.020089  -3.844 0.000121 ***
a[m.PTGENDERFemale]  0.266786   0.007268  36.705  < 2e-16 ***
a[zEduc]             0.210118   0.021580   9.737  < 2e-16 ***
b[(Intercept)]       0.842972   0.015688  53.734  < 2e-16 ***
b[zwb]              -0.589807         NA      NA       NA
b[zAge]             -0.124395   0.033468  -3.717 0.000202 ***
b[m.PTGENDERFemale]  0.600601   0.040893  14.687  < 2e-16 ***
b[zEduc]             0.122145   0.034828   3.507 0.000453 ***
w[(Intercept)]      -0.426748   0.014152 -30.154  < 2e-16 ***
w[drcs]              1.169788   0.008142 143.666  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1



  Null deviance: 3895.6  on 3317  degrees of freedom
Linear deviance: 2026.9  on 3312  degrees of freedom
Logist deviance:     NA  on   NA  degrees of freedom
 Delay deviance: 2475.7  on 3306  degrees of freedom

             R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.2880251 -3890.378    6 7792.755 7792.781 7829.398
Cusp model   0.3869597 -3622.986   12 7269.971 7270.066 7343.257
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 534.8, df = 6, p-value = 0
```

```
Number of optimization iterations: 43
```

Pesimo para el AV45

```
> summary(fit)

Call:
cusp(formula = y ~ drcs, alpha = alpha ~ zwb + zAge + m.PTGENDER +
    zEduc, beta = beta ~ zwb + zAge + m.PTGENDER + zEduc, data = datac)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.77918  -0.53934  -0.09885   0.52564   2.93425

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]      -0.329897         NA      NA       NA
a[zwb]              -0.509019   0.029985 -16.976  < 2e-16 ***
a[zAge]             -0.175968   0.022751  -7.735 1.04e-14 ***
a[m.PTGENDERFemale]  0.492413   0.050822   9.689  < 2e-16 ***
a[zEduc]             0.185683   0.026500   7.007 2.44e-12 ***
b[(Intercept)]       0.149342         NA      NA       NA
b[zwb]               0.169838   0.006714  25.295  < 2e-16 ***
b[zAge]             -0.183744         NA      NA       NA
b[m.PTGENDERFemale]  0.701522         NA      NA       NA
b[zEduc]             0.127566         NA      NA       NA
w[(Intercept)]      -0.049160   0.016707  -2.943  0.00326 **
w[drcs]              1.088235   0.010977  99.138  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


  Null deviance: 1649.6  on 1691  degrees of freedom
Linear deviance: 1051.9  on 1686  degrees of freedom
Logist deviance:     NA  on   NA  degrees of freedom
 Delay deviance: 1403.4  on 1680  degrees of freedom

             R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.2448373 -1998.714    6 4009.428 4009.478 4042.030
Cusp model   0.1503419 -2015.754   12 4055.508 4055.693 4120.712
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 34.08, df = 6, p-value = 6.494e-06
```

```
Number of optimization iterations: 68
```

Pero bastante bueno para el PiB

```
> summary(fit)

Call:
cusp(formula = y ~ drcs, alpha = alpha ~ zwb + zAge + m.PTGENDER +
    zEduc, beta = beta ~ zwb + zAge + m.PTGENDER + zEduc, data = datac)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.39747  -0.34765   0.02365   0.61995   2.80434

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
a[(Intercept)]      -0.41926    0.10606  -3.953 7.71e-05 ***
a[zwb]              -0.41509    0.08006  -5.185 2.16e-07 ***
a[zAge]             -0.01542    0.06966  -0.221  0.82483
a[m.PTGENDERFemale]  0.01350    0.14087   0.096  0.92365
a[zEduc]             0.15078    0.07280   2.071  0.03835 *
b[(Intercept)]       1.38998    0.24548   5.662 1.49e-08 ***
b[zwb]               0.09861    0.13536   0.729  0.46629
b[zAge]             -0.33692    0.11845  -2.844  0.00445 **
b[m.PTGENDERFemale]  0.61578    0.24049   2.561  0.01045 *
b[zEduc]             0.04128    0.12358   0.334  0.73837
w[(Intercept)]      -0.49934    0.05477  -9.116  < 2e-16 ***
w[drcs]              1.23598    0.04958  24.928  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


  Null deviance: 297.20  on 217  degrees of freedom
Linear deviance: 154.46  on 212  degrees of freedom
Logist deviance:     NA  on  NA  degrees of freedom
 Delay deviance: 152.70  on 206  degrees of freedom

             R.Squared    logLik npar      AIC     AICc      BIC
Linear model 0.2060384 -271.7741    6 555.5482 555.9463 575.8552
Cusp model   0.5180603 -235.4183   12 494.8367 496.3586 535.4506
---
Note: R.Squared for cusp model is Cobb's pseudo-R^2. This value
      can become negative.

    Chi-square test of linear vs. cusp model

X-squared = 72.71, df = 6, p-value = 1.135e-13
```

```
Number of optimization iterations: 34
```