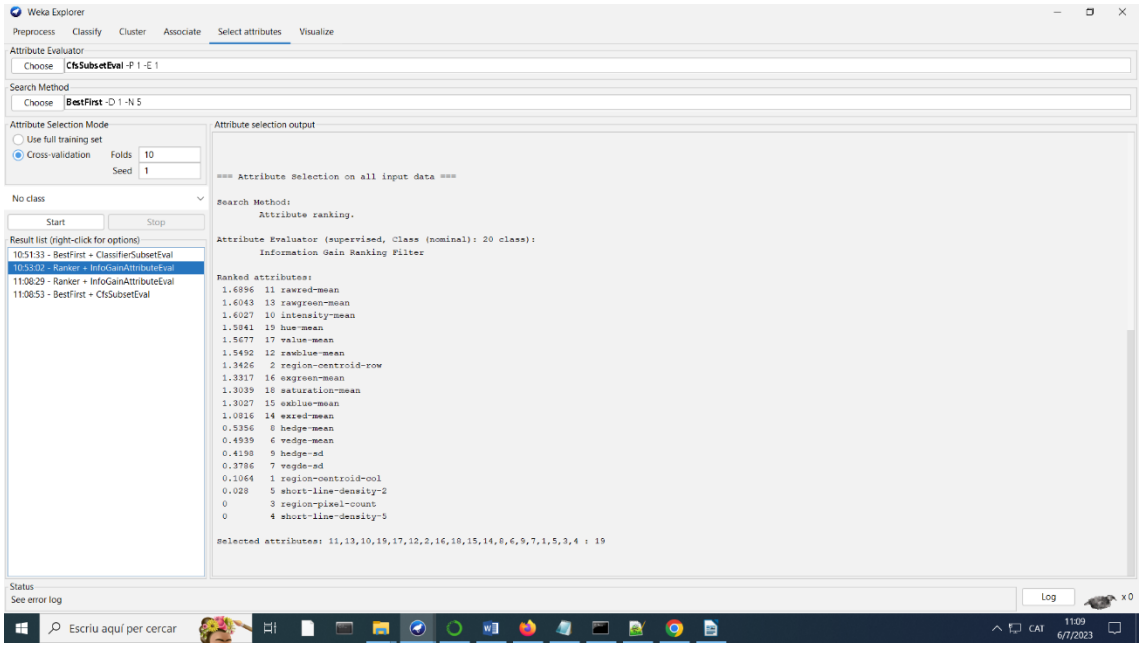
Pràctica final. Weka

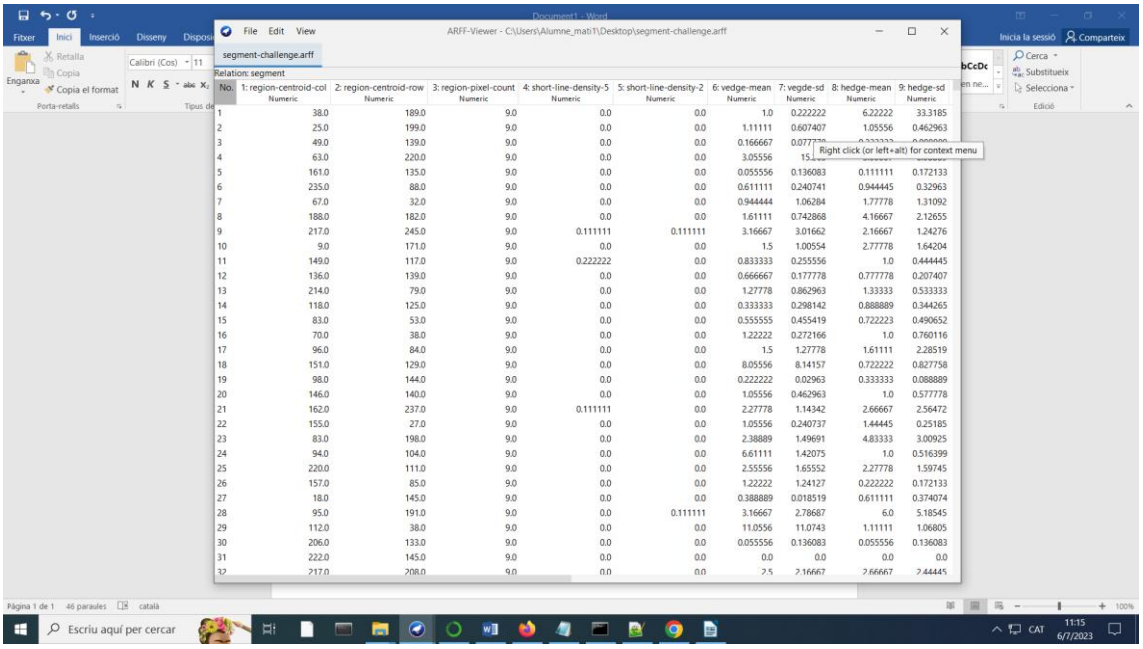
Primer valoro quins atributs son menys rellevants per poder eliminar-los i que la meva predicció s'acosti millor al valor real.



Amb aquesta

Amb aquesta comprovació decideixo eliminar les dues darreres columnes que ens indica en la imatge de dalt que tenen menys incidència.

En la següent imatge també comprovem que la influencia es poca i puntual.



Iniciem el primer model predictiu:

NaiveBayes:

Scheme: weka.classifiers.bayes.NaiveBayes

Relation: segment-weka.filters.unsupervised.attribute.Remove-R4-5

Instances: 1500

Test mode: split 80.0% train, remainder test

Time taken to build model: 0 seconds

=== Evaluation on test split ===

Time taken to test model on test split: 0 seconds

=== Summary ===

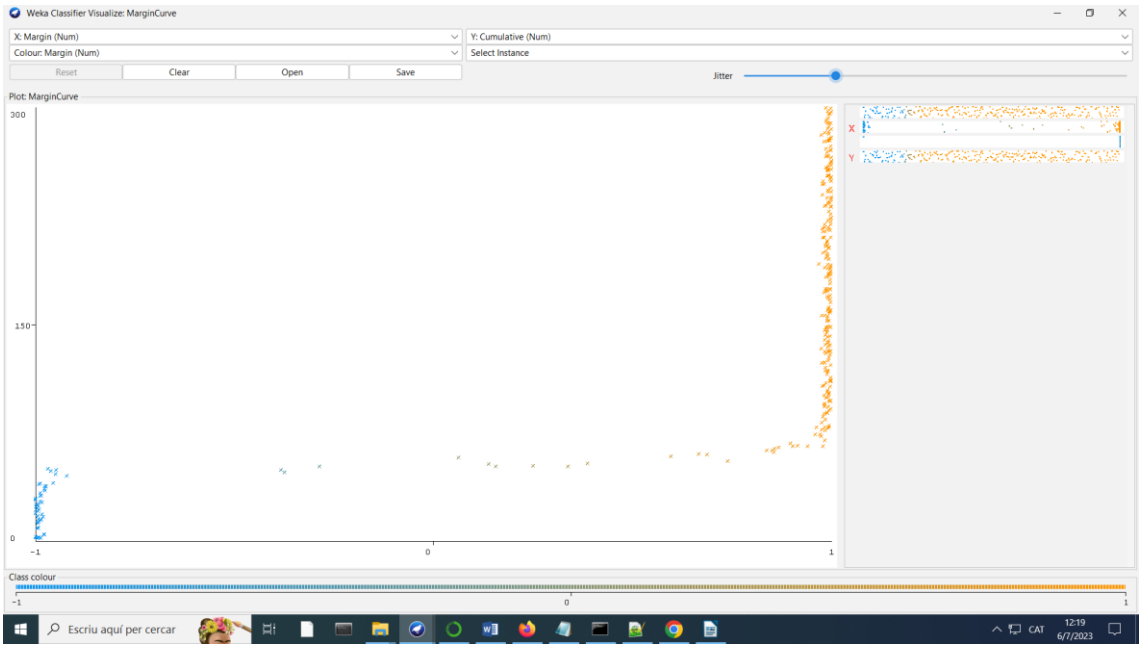
Correctly Classified Instances	250	83.3333 %
Incorrectly Classified Instances	50	16.6667 %

=== Detailed Accuracy By Class ===

Weighted Avg.	0,833	0,023	0,847	0,833	0,816	0,807	0,977	0,897

Decideixo fer servir percentatge Split 80% perquè el procés és més ràpid.

Aquest model és el pitjor dels tres escollits perquè en el Summary els valors escollits (a dalt) son més baixos que en els altres dos, a més Accuracy també doma un resultat més baix.



Logistic:

Scheme: weka.classifiers.functions.Logistic -R 1.0E-8 -M -1 -num-decimal-places 4

Relation: segment-weka.filters.unsupervised.attribute.Remove-R4-5

Instances: 1500

Test mode: split 80.0% train, remainder test

Time taken to build model: 0.58 seconds

=== Evaluation on test split ===

Time taken to test model on test split: 0 seconds

=== Summary ===

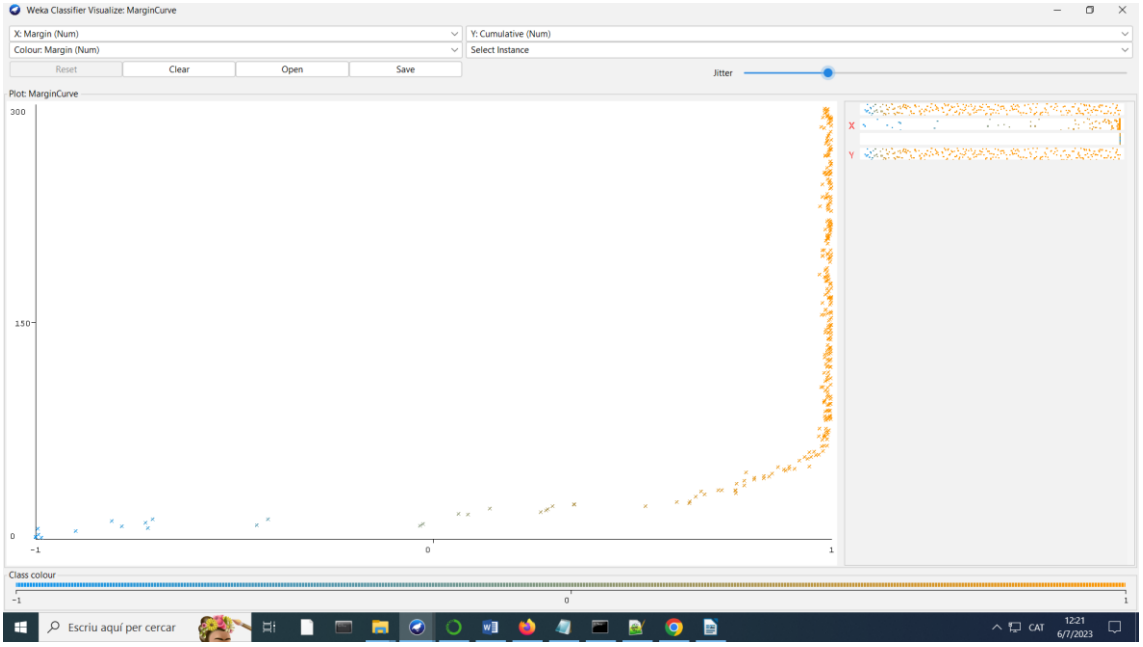
Correctly Classified Instances	286	95.3333 %
Incorrectly Classified Instances	14	4.6667 %

=== Detailed Accuracy By Class ===

Weighted Avg.	0,953	0,007	0,956	0,953	0,954	0,947	0,996	0,981

Escullo en aquest cas el percentatge Split per la mateixa raó que abans, el temps del procés del model és més curt.

Aquest model s'acosta més a unes dades predictives més valides amb valors de les instancies correctes i incorrectes més fiables. També noto millora en la precisió. Ens acostem a un model millor d'acord amb les dades.



RandomForest:

Scheme: weka.classifiers.trees.RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1

Relation: segment-weka.filters.unsupervised.attribute.Remove-R4-5

Instances: 1500

Test mode: split 80.0% train, remainder test

=== Evaluation on test split ===

Time taken to test model on test split: 0.01 seconds

=== Summary ===

Correctly Classified Instances	298	99.3333 %
Incorrectly Classified Instances	2	0.6667 %

=== Detailed Accuracy By Class ===

Weighted Avg.	0,993	0,001	0,994	0,993	0,993	0,992	1,000	0,997

Amb la mateixa raó que els altres dos models escullo la modalitat del test de 80%/20% perquè és més rapida en la execució.

Comprovant les dades determino que aquest model RandomForest és el que millors dades predictives ens donarà. Podem comprovar que els valors son més alts en tots els valors presentats en aquest document.

He escollit sempre els mateixos tipus de valors per poder fer una comparació més fiable, tot hi que cada model dona més importància i informació sobre valors diferents (com pot ser el cas de les mitjanes en el model NaiveBayes)

Finalment he posat les imatges de la MarginCurbe on es veu que la NiveBayes és la mes fluixa i com els altres dos models s'acosten més a una millor predicció.