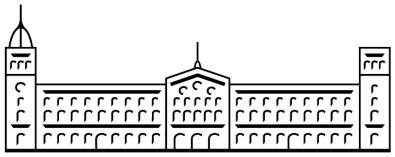


Treball final de grau

GRAU D'ENGINYERIA INFORMÀTICA

Facultat de Matemàtiques
Universitat de Barcelona

Ciència de les dades aplicada a resultats acadèmics

Autor: Xavier Moreno Liceras

Directora: Laura Igual
Realitzat a: Departament
Matemàtica Aplicada y Anàlisis

Barcelona, June 22, 2015

Ha de ser redactat en primer persona del plural del present

Revisió de les faltes d'ortografia

Abstract

Fins ara a la Facultat de Matemàtiques de la Universitat de Barcelona, un tutor d'estudis té assignat un grup d'estudiants. Aquest no pot tenir un coneixement ampli de la situació de cada alumne i és per això que aplica una serie d'accions comunes per a cadascún.
Aquest treball forma part d'un projecte d'innovació docent, en el que proposem una eina de suport per al tutor d'estudis, la qual permeti ajudar al tutor a conèixer millor el perfil de cada alumne que tutoritza, amb el suport de dades estadístiques per perfil d'alumne i un recomanador per determinar la dificultat del curs que li pot costar un alumne.

Resum

Fins ara a la Facultat de Matemàtiques de la Universitat de Barcelona, un tutor d'estudis té assignat un grup d'estudiants. Aquest no pot tenir un coneixement ampli de la situació de cada alumne i és per això que aplica una serie d'accions comunes per a cadascún. Aquest treball forma part d'un projecte d'innovació docent, en el que proposem una eina de suport per al tutor d'estudis, la qual permeti ajudar al tutor a conèixer millor el perfil de cada alumne que tutoritza, amb el suport de dades estadístiques per perfil d'alumne i un recomanador per determinar la dificultat del curs que li pot costar un alumne.

Resumen

Fins ara a la Facultat de Matemàtiques de la Universitat de Barcelona, un tutor d'estudis té assignat un grup d'estudiants. Aquest no pot tenir un coneixement ampli de la situació de cada alumne i és per això que aplica una serie d'accions comunes per a cadascún. Aquest treball forma part d'un projecte d'innovació docent, en el que proposem una eina de suport per al tutor d'estudis, la qual permeti ajudar al tutor a conèixer millor el perfil de cada alumne que tutoritza, amb el suport de dades estadístiques per perfil d'alumne i un recomanador per determinar la dificultat del curs que li pot costar un alumne.

Agraïments

Vull agrair a ...

Contents

1 Introducció

Un dels components bàsics de l'activitat docent és l'acció tutorial, la qual té com a finalitat guiar i aconsellar a l'estudiant durant la seva etapa d'estudis. Ajuda a l'estudiant a millorar el seu rendiment, a la seva orientació professional, i el més important, ajudar a prendre decissions que afavoreixin els seus estudis i la seva satisfacció. Per un altre banda tenim el pla d'acció tutorial (PAT) que tracta d'un document amb un conjunt ordenat d'accions sistemàtiques previament planificades. Una de les coses que impulsa el PAT és l'assignació d'un tutor d'estudis a un grup d'estudiants. Un tutor d'estudis, per tant té com a finalitat entre altres, acompanyar a l'alumnat durant el seu transcurs estudiantil des de l'inici del grau fins al final, donant consell per cara al món professional.

Ens hem trobat amb el problema que un tutor d'estudis com tutoritza a un grup d'alumnes no és capaç de contemplar detingudament cadascun dels seus alumnes. Els pot guiar de forma genèrica, així seguint el pla d'acció tutorial. S'ha pogut observar al llarg dels anys, per exemple, que alumnes amb qualificacions moderades a primer i segon del grau d'Enginyeria Informàtica tenen problemes per afrontar certes assignatures de tercer. Ara bé, això s'ha pogut observar al llarg dels anys, però i si estem evitant altres problemes o fets que no s'han pogut obervar fins ara? És això el que volem explorar i fer conclusions que no s'hagin pogut arribar. Arran de tot això, es va fer una petició al Vicerectorat de Política Docent per dur a terme un projecte que facilités el treball al tutor d'estudis. D'aquí va nèixer el projecte com un projecte d'innovació docent, amb el títol de: *Sistema intel·ligent de suport per al tutor d'estudis.*

La finalitat del projecte d'innovació docent és la creació d'una eina que el tutor pogui consultar i li ajudi a prendre decissions cara a les seves tutories. Aquesta eina ha de permetre al tutor visualitzar la trajectoria d'un alumne, fer recomanacions específiques per cadascun d'ells, entre altres. Un dels recursos principals d'aquest projecte són les dades, ja que són la que ens permetran arribar a conclusions i poder construïr l'eina per al tutor d'estudis. Les dades han sigut obtingudes a través del Vicerectorat de Política Docent.

Aquest treball forma part del projecte d'innovació docent, i ens centrem en l'estudi estadístic dels resultats acadèmics de la Facultat de Matemàtiques de la UB. L'objectiu és montar una base per poder montar en la següent fase del projecte el sistema per al tutor d'estudis. En aquest treball ens hem cen-

trat en l'exploració dels perfils d'estudiants dels primers cursos de cadascun dels graus impartits en la Facultat de Matemàtiques. També s'ha treballat en un sistema de predicció de notes d'un alumne cara a la seva próxima matriculació. A més s'ha desenvolupat un ranking de dificultat de notes no matriculades d'un alumne a partir del predictor de notes. L'objectiu general d'aquest treball és obtenir coneixement a partir de les dades, i és per això que hem convertit aquest projecte en un projecte de ciència de les dades.

2 Descripció del problema

2.1 Projecte d'innovació docent

Aquest treball de fi de grau, ve arran d'un projecte d'innovació docent que va nèixer del Departament de Matemàtica Aplicada i Anàlisi (MAIA) i Departament de Metòdes de Investigació i Diagnòstic en Educació (MIDE). El participants del projecte són:

- Dra. Laura Igual Muñoz (MAIA)
- Dr. Santiago Seguí Mesquida (MAIA)
- Dr. Eloi Puertas Prats (MAIA)
- Dr. Oriol Pujol (MAIA)
- Dr. Jordi Vitrià Marca (MAIA)
- Dra. Petia Radeva (MAIA)
- Dr. Luís Garrido Ostermann (MAIA)
- Dra. Maria del Pilar Folgueiras Bertomeu (MIDE)

Com ja s'ha dit, la finalitat del projecte d'innovació docent és el desenvolupament d'un sistema intel·ligent de suport al tutor d'estudis, i per dur-lo a terme el projecte s'ha dividit en 5 fases.

Fase 1 Adquisició, ordenació, centralització i anonimització de les dades curriculars disponibles dels alumnes. La fase inicial on es deixen les dades preparades per poder treballar amb elles.

Fase 2 Anàlisi de les dades mitjançant tècniques de ciències de les dades. Fer un anàlisi estadístic de les dades que tenim i aplicar ciència de les dades per explorar la informació amagada darrere de les dades.

Fase 3 Anàlisi de les dades mitjançant tècniques d'aprenentatge automàtic. A partir de les dades aplicar algoritmes de predicció de dades per poder predir les notes d'un alumne en base a les seves notes i la de la resta.

Fase 4 Desenvolupament del sistema intel·ligent. En aquesta fase es busca el desenvolupament de la eina de suport per al tutor d'estudis.

Fase 5 Avaluació. S'avalua el sistema per tal de fer proves, i buscar mancances i errors del propi sistema.

Aquest treball forma part de la fase 1, 2 i 3. Les fases 4 i 5 són l'altre part del projecte, i no s'en parlarà en aquest treball.

2.2 Ciència de les dades

La ciència de les dades és el conjunt d'etapes per tal d'arribar a un resultat, en forma de coneixement, a partir d'un conjunt de dades. Aquesta aplica un conjunt de tècniques de diferents àreas, ara com matemàtiques, estadística, teoria de la informació o tecnologia de l'extracció d'informació.

Un projecte de ciència de les dades es separa en diverses etapes:

1. **Plantejament de preguntes** Què és el que volem explorar? Té sentit el que ens estem plantejant?
2. **Adquisició de les dades** Com és la font d'obtenció de les dades? (Base de dades, *Web Scraping*, fitxer .csv)
3. **Descripció** Aquesta fase abasta tres processos
   (a) **Neteja de dades** Com hem de netejar i separar les dades? (mostres atípiques, filtració, redució de dimensions, normalització, extracció de característiques)
   (b) **Agregació** Com hem de recolectar i resumir les dades? (promig, desviació estàndard, box plots)
   (c) **Enriquiment** Com podem afegir més informació a les nostres dades? (Cerca a altres fonts de dades adicionals)
4. **Descobriment** Podem segmentar les nostres dades per trobar grups naturals i disgregats? (Clusterització, visualització)
5. **Anàlisis** Com hem de modelar les nostres dades? (Com estan de relacionades cada variable?, Com podem determinar quines són les variables importants?)
6. **Predicció** A partir de les dades que tenim, que podem predir del futur? (Regresions, clasificadors, recomanadors)
7. **Evaluació** Com de segur estem dels nostres resultats? (Proves estadístiques, rendiment del model)

2.3 Etapes del projecte

2.3.1 Plantejament de preguntes

La primera etapa és el plantejament de les preguntes que voliem resoldre. A partir de la plataforma trello (explicada en la secció d'eines), entre els participants del projecte vam plantejar preguntes, les quals entre tots decidiem amb quines preguntes ens quedariem i respondriem. Moltes de les preguntes no podíem saber si les podíem respondre fins que ens arribessin les dades, ja que depeniem totalment de la informació que contenien aquestes.

2.3.2 Adquisició

L'adquisició de les dades va ser a partir del Vicerectorat de Política Docent. Aquest ens va proporcionar les dades a través d'una fulla de càlcul. Tot i que les dades vinguessin anonimitzades i tractades pel departament corresponent, vam haver de fer una neteja de dades.

2.3.3 Neteja de dades

En aquesta etapa hem hagut de netejar les dades per tal de poder treballar amb elles. Aquestes van ser les netejes que es van fer:

Canvi de format Com ja s'ha explicat anteriorment les dades van arribar en un full de càlcul, on en cada fulla havia una taula amb diferent informació. Per poder manipular-les millor des de Python, es va separar cada fulla en un fitxer amb format *csv*, de tal manera que va quedar un fitxer *csv* per taula. En la próxima secció s'explica amb detall les dades obtingudes.

Canvi de nom de les columnes Per tal de poder creuar les diferents taules, els noms de les columnes havien de ser el mateix.

Enriquiment de les dades A partir d'una font externa hem pogut adquirir el curs i semestre que es cursa cada assignatura, per tant el que fem és creuar aquestes dades amb les dades que tenim de cada assignatura per tal de tenir més informació per assignatura.

Unió de graus L'any 2009 el grau en Enginyeria Informàtica de la UB tenia com a codi *G1041*, però a partir de l'any 2010 el codi va passar a ser *G1077*. Les assignatures eren les mateixes, tot i que tenien codis diferents també. Vam procedir a fer la unió dels *G1041* amb *G1077*, per tal de no perdre informació rellevant, ni considerar-la per separat.

Eliminació del curs 2014, segon semestre Explorant les dades em vaig adonar que alumnes que s'havien matriculat l'any 2014, però encara no havien acabat de cursar l'assignatura, en aquesta els hi apareix un 0. Això fa que dintre de les notes dels alumnes hi hagin dades incoherents, per aquesta raó es va decidir eliminar totes les notes del segon semestre i de l'any 2014. El percentatge d'eliminació de dades és d'un 10.91% del total de notes que tenim.

Normalitació de les notes Per tal d'evitar els canvis de mitja i variança en cada assignatura cursada per any, ja sigui per un canvi de professor, canvi de pla docent, diferents promocions, ...es va decidir normalitzar les notes per any i per assignatura aplicant una normalització d'unitat tipificada en la qual s'aplica per cada dada la següent fórmula:

$$z = \frac{x - \mu}{\sigma},$$

on μ és el promig per any i per assignatura, i σ és la desviació estàndard per any i per assignatura. Amb això aconseguim mitja 0 i desviació estàndard 1.

2.3.4 Clusterització

Aquesta etapa era necessaria per poder respondre a una des les preguntes plantejades, i és: *Hi ha diferents perfils d'alumnes?*. Per tal de respondre a aquesta pregunta s'han aplicat mètodes de clusterització a partir de les notes dels alumnes diferenciats per cursos.

2.3.5 Predicció

La predicció, com s'ha explicat abans, és la predicció del futur a partir de les dades disponibles. En aquest cas hem volgut predir les notes que pot arribar a treure un alumne en base a les notes que ha tret en cursos anteriors.

2.3.6 Evaluació

Un cop construida la predicció, hem d'avaluar quant de bona és. S'ha d'evaluar de forma quantitativa (mitjançant mètriques) i qualitativament (amb la mostra de casos).

2.4 Explicació de les dades obtingudes

En aquest apartat s'explicarà la informació més rellevant que podem trobar en les nostres dades. Recordem que les dades ens havien arrivat en una fulla de càlcul, i aquesta l'hem separat per diversos fitxers amb format .csv.

Les dades que hem pogut adquirir són molt enriquidores, tenen la informació necesaria per fer un estudi ampli tant per als estudiants com per a l'estudi d'assignatures. A més les dades venen anonimitzades, a priori no podem obtenir la informació d'un alumne que coneguem. Les dades les tenim separades en diferents fitxers, els quals estan relacionats entre si mitjantçant identificadors, com ara un identificador d'alumne o el codi d'una assignatura. Els fitxers són els següents:

Informació general de l'estudiant Aquest fitxer conté per cada fila informació sobre un alumne en termes de matriculació: l'any d'inici de carrera, grau que realitza, la via amb la qual va accedir a la carrera, la nota d'accés a la Universitat, entre altres.

Informació d'assignatures Aquí podem trobar la informació de cada assignatura que existeix en els graus d'Enginyeria Informàtica i Matemàtiques. Per cada fila tenim la següent informació: l'identificador de l'assignatura, el nom de l'assignatura, els crèdits ECTS corresponents a aquesta i el grau a la que pertanyen. A més a més, vam obtenir mitjantçant una altre font d'informació, per cada assignatura de quin curs i semestre es tractava. Aquesta dada la vam creuar amb l'anterior per ampliar la informació per assignatura.

Qualificacions per alumne i per assignatura Per últim i més important, el fitxer que conté les qualificacions de tots els alumnes per assignatura, és a dir, per cada fila tenim: l'identificador de l'alumne que realitze l'assignatura, l'identificador de l'assignatura realitzada, la qualificació d'aquella assignatura, l'ensenyament del qual es tracta, l'any en el que es va realitzar l'assignatura i el tipus d'apunt (ordinaria, reconeixement o convalidada).

2.5 Preguntes plantejades

Hi ha diferents perfils d'alumnes?

A partir de la distribució de les notes de cada alumne per cada assignatura que ha fet, podem determinar que hi ha diferents perfils d'estudiants? Això és el que ens estem preguntant. S'han agafat tots els alumnes que hagin cursat totes les assignatures de primer i després les de segon, tant al grau d'Enginyeria Informàtica com al grau de Matemàtiques. La experiència ens diu que hi han alumnes bons en programació i dolents en les assignatures de matemàtiques a primer del grau d'Enginyeria Informàtica. Però per a la resta de cursos, quins perfil podem trobar? Ara que tenim les dades això ho podem saber, convertirem les dades en coneixement.

Quina es la tassa d'abandonament per cada tipus de perfil?

A partir dels perfils que han sigut determinats en la pregunta anterior, quin és el percentatge d'abandonament per cadascun d'aquests. Volem saber si és cert que els alumnes que van a parar al perfil d'alumnes que ho suspenen tot són els que solen abandonar la carrera. Fins ara això és el que podem saber a partir de l'experiència, però es pot demostrar amb dades i corroborar-ho.

Cadascun d'aquests perfils amb quin perfil de provinença encaixa?

Al llarg dels anys s'ha pogut notar que els alumnes que solen treure bones notes a primer d'Enginyeria Informàtica, acaben treien bones notes a segon la gran majoria. Ara bé, això és cert? Per això ens plantegem aquesta pregunta, a partir de perfils d'origen, volem saber amb quin perfil de destí solen anar. En aquest cas hem fet els següents creuaments per cada grau:

Origen	Destí
Via d'accés	Perfil d'alumnes de primer
Perfil d'alumnes de primer	Perfil d'alumnes de segon

Els perfils d'alumnes de primer i segon són els perfils determinats a la primera pregunta, i els perfils de via d'accés que hem seleccionat han sigut els següents:

1. Batxillerat (Batx)
2. Salt d'Universitat (Uni)

Hem obviat els alumnes provinents de cicle o ja diplomants, per la seva baixa presència. Les vies d'accés només les comprovem amb els alumnes que hagin cursat totes les assignatures de primer, sense convalidar, el que fa que no apareguin una gran quantitat d'alumnes de cicle, ja que la majoria d'aquests tenen alguna assignatura convalidada de primer. I pel que fa als alumnes ja diplomats, no hi han masses en totes les dades presents.

Predicció de notes i ranking de dificultat d'assignatures

A partir de les notes que ha tret un alumne en el seu passat, podem predir quines assignatures li aniran bé i malament en el futur? Bé, això és el que ens plantejem en aquesta última pregunta, volem recomanar a un alumne en quines assignatures no li aniran gaire bé per a que així pogui reforçar més el temari que es donarà en aquella assignatura. Recordem que la finalitat del projecte d'innovació docent és que aquesta eina sigui un suport per al tutor, és a dir, la predicció no ens dirà el que ha de fer un alumne, aquesta decisió es delega al tutor que a partir de l'eina decidirà que fer.

3 Planificació

3.1 Tasques

Les tasques d'aquest projecte són semblants a les etapaes d'un projecte de ciència de les dades. Les tasques són:

- Formació
- Plantejament de preguntes
- Neteja de dades
- Clusterització
- Predicció
- Evaluació
- Documentació

Les úniques etapes noves que trobem són: la de formació, és la etapa dedicada l'aprenentatge autònom de les eines utilitzades; la de documentació, és el període de temps per tal de desenvolupar aquesta memòria.

3.2 Diagrama de Gantt

S'ha construït dos diagrames de Gantt, un a partir de la planificació inicial i l'altre amb la planificació real per tal de veure les diferències.

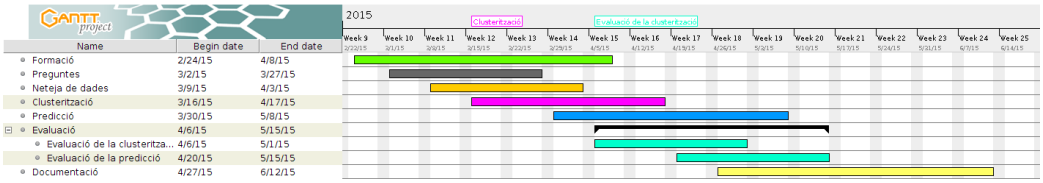


Figure 1: Planificació inicial

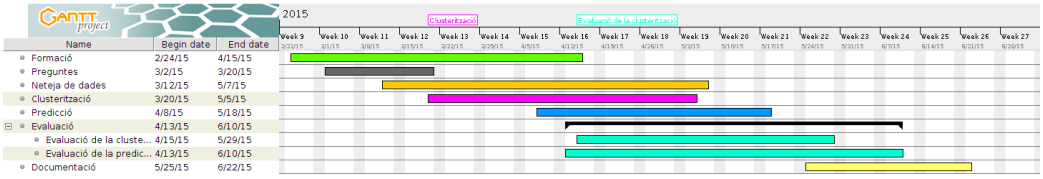


Figure 2: Planificació real

El Treball de fi de Grau equival a 18 crèdits ECTS, si cada crèdit equival a 25 hores, llavors tenim:

$$18\,crèdits \cdot \frac{25\,hores}{1\,crèdits} = 450\,hores$$

Per tant totes les tasques s'han de dividir en 450 hores, les hores dedicades han sigut les següents:

	Formació	Preguntes	Neteja de dades	Clusterització	Predicció	Evaluació	Documentació
Hores	25	25	50	75	75	125	50

Table 1: Hores de dedicació per cada tasca

3.3 Evaluació económica

	Hores	Preu per hora (Euro)	Preu total (Euro)
Formació	25	0	0
Plantejament de preguntes	25	10	250
Neteja de dades	50	20	1000
Clusterització	75	25	1875
Predicció	75	25	1875
Evaluació	125	25	3125
Documentació	75	0	0
TOTAL	450		8125

Table 2: Taula d'evaluació económica

El projecte sortiria per 8125 euros, en els quals s'inclou en la etapa de Evaluació, una documentació dels resultats obtinguts i les conclusions d'aquests.

4 Desenvolupament del projecte

4.1 Eines

4.1.1 Eines de suport

Aquestes són les eines de suport que ens han ajudat al llarg del treball per tal de fer més còmode la seva organització tant personal com per equip.

GitHub

GitHub és una plataforma online per desenvolupar projectes software de forma col·laborativa. Aquesta plataforma utilitza un control de versions anomenat Git. La finalitat de GitHub és l'emmagatzenament massiu de projectes amb codi font obert. Per això hem optat per la utilització de GitHub, ja que que volem que el nostre codi el pogui veure tothom i que qualsevol que el necessiti per fer la seva investigació, el pogui utilitzar.

Bitbucket

Bitbucket és una plataforma semblant a GitHub, però amb el servei d'un altre control de versions com Mercurial a més de Git. Bitbucket té l'advantatge de permetre crear repositoris privats de forma gratuïta. Aquesta plataforma va bé per a l'inici d'un projecte on es fan molts canvis en el codi, ja que pots tenir el codi en privat, i un cop el codi ja agafa forma es pot migrar a GitHub. Això és el que hem fet nosaltres en el projecte, començar amb Bitbucket i després passar-nos a GitHub amb el codi font obert.

Trello

Per últim com eina de suport, hem fet servir Trello, una plataforma online que permet una comunicació més clara entre els membres d'un projecte. Amb Trello pots crear projectes i cada projecte conté un conjunt de llistes que s'omplen de tasques. Hem fet servir Trello per comunicar-nos amb la tutora i tenir present una planificació per tal d'organitzar-nos millor.

4.1.2 Eines de programació

En aquesta secció trobarem amb el llenguatge de programació, i conjunt de llibreries, que hem treballat.

Python

Python és un llenguatge d'alt nivell interpretat. Remarquen molt la fàcil lectura del seus codis, per això té una sintaxis molt semblant a un pseudocodi. Python és un llenguatge de codi obert i desenvolupat per *Python Software Foundation*, una organització sense ànim de lucre. Vam escollir Python per dues raons: per ser un llenguatge de scripting i per la seves llibreries relacionades amb el tractament de dades (com Pandas, NumPy o Scikit-learn).

Pandas

Pandas és una biblioteca informàtica escrita en Python per a la manipulació i anàlisi de dades. Especialment va bé per al tractament de taules alhora de fer consultes, o per a l'agrupació i agrecació d'informació.

NumPy

Numpy és una biblioteca informàtica de Python per operar amb vectors i matrius d'una forma més extensa a la que et permet el mateix llenguatge Python, la qual conté tot un conjunt de funcions matemàtiques d'alt nivell per treballar amb aquests vectors i matrius.

Scikit-learn

Scikit-learn (o sklearn) és una biblioteca informàtica orientada a l'aprenentatge automàtic per a Python. Té suport per classificadors, regressors i clusterització. Per aquest projecte hem fet servir clustering i regressors. En la secció de Tècniques utilitzades es detalla cada tècnica utilitzada d'aquesta biblioteca informàtica.

Bokeh

Bokeh és una biblioteca informàtica per a la visualització interactiva de dades dirigida als navegadors per a la seva presentació a través d'HTML i JavaScript. Bokeh té el suport per a gràfiques específiques com diagrames de barra, box plots o time series, però a banda d'aquests gràfics pots dibuixar sobre un gràfic amb elements bàsics com cercles, línies, rectangles, entre altres.

Seaborn

Per últim tenim Seaborn que també és una biblioteca informàtica per a visualitazció de dades com Bokeh, amb gràfiques molt més específiques. A més té una part de la biblioteca informàtica dedicada a les paletes de colors i la qual permet escollir un conjunt de colors afavorits per mostrar les dades.

```
%matplotlib inline
import seaborn as sbn
palette = sbn.color_palette("hls", 5)
sbn.palplot(palette)
```

Figure 3: Elecció d'una paleta de 5 colors

4.1.3 Eines d'edició

IPython notebook

Ipython notebook és un editor per a l'entorn de Python. La filosofia *notebook* s'emprea per tenir un codi molt més llegible i a més tenir explicacions d'allò que es programa, ja que es pot barrejar codi, la sortida del codi, markdown, HTML, entre altres. Hem optat per escollit aquest entorn d'edició ja que en un projecte de ciència de les dades s'han de veure resutats constants i poder-los comentar.

Texmaker

Aquesta és una eina d'edició de LaTeX, la qual permet poder generar informes, documents, llibres d'una forma més programàtica. A partir d'un etiquetatge estipulat es poden generar documents amb un estil predefinit com el d'aquesta memòria.

4.2 Tècniques utilitzades

4.2.1 Clusterització (Agrupacions)

La clusterització és molt important en el món de les dades, permet reconèixer diferents grups de ítems de les nostres dades, en el nostre cas d'alumnes. Per això, abans de veure els resultats i experiments explorats, cal entendre les diferències entre les diferents tècniques de clusterització. En aquest projecte hem fet servir dues tècniques, on l'objectiu d'elles és el mateix, desframentar les dades i trobar diferents grups d'alumnes. Aquestes dues tècniquies són K-means i MeanShift, ambdues implementades en la biblioteca informàtica de Scikit-learn. link

K-means

K-means probablement és un dels algoritmes d'agrupació més conegut. Partint de n elements, divideix aquests n elements en k grups (argument obligatòri de l'algoritme) on cada element pertany al grup més proper a la mitjana. L'algoritme de *K-means* està descrit per la següent fórmula: Referenciar al k-means de sklearn.

Tenint un conjunt d'elements $(x_1, x_2, \ldots, x_n)$ on cada elements és un vector d dimensional, *K-means* construeix una partició dels elements en k grups, on $k \leq n$ quedant $S = \{S_1, S_2, \ldots, S_k\}$. Amb la finalitat de minimitzar la suma dels quadrats dintre de cada grup:

$$\underset{S}{arg\,min} \sum_{i=1}^{k} \sum_{x_j \in S_i} \|x_j - \mu_i\|^2$$

on μ_i és el centroide dels punts del conjunt S_i, és a dir, el punt mig.

Com es veu en la fòrmula, aquest algoritme depén d'una k, per determinar agrupacions, per tant *K-means* ha de rebre com paràmetre d'entrada quants grups busquem. També podem pensar que depén del centroide μ_i, però no es necessari, ja que aquest convergeix si s'apliquen x iteracions sobre la fòrmula.

Mean Shift

Mean Shift és l'altre tècnica d'agrupació o clusterització que utilitzo en aquest projecte. L'objectiu d'aquesta tècnica és el mateix que *K-means*, però el seu algoritme funciona de forma diferent, considerant l'espai de característiques com una funció de densitat de probabilitat.

Referenciar al mean shift de sklearn.

Aquest algoritme no necessita com a entrada el número de clusters que busquem, com *K-means*. Té altres paràmetres d'entrada, però són opcionals. En aquesta imatge podem veure la diferència entre *K-means* i *Mean Shift*.

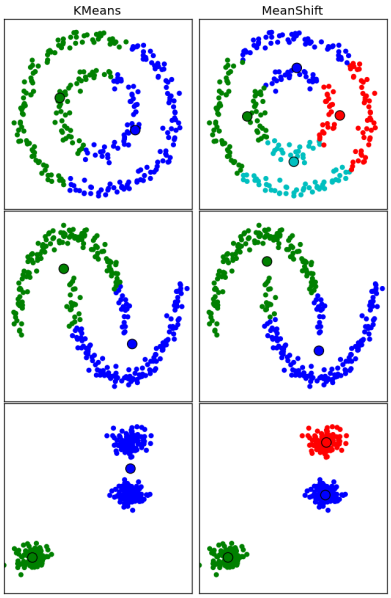


Figure 4: Comparació de K-means amb Mean Shift

Com es veu en la figura d'adalt, en el tercer gràfic de cada tècnica, tenim que amb *MeanShift*, com no hem de dir el número de divisions que volem, ell mateix ens diu que hi han tres grups. Però amb *K-means* si possem dos clusters, per exemple, estariem unificant dos clusters en un.

Mètriques utilitzades

Existeixen dos indicadors d'avaluació dels resultats de l'anàlisis:

1. **Supervisat** Utilitza les agrupacions reals per comparar-les amb les agrupacions donades per l'algoritme de clusterització.
2. **No supervisat** És tot lo contrari, mesura la qualitat del propi model, basant-se en les característiques d'aquest.

En el nostre cas, el que volem és explorar i averiguar quins perfils d'estudiants hi han, per tant hem d'utilitzar mètriques no supervisades, ja que no tenim una referència per comparar. La única mètrica no supervisada que utilitzem és la *Silhouette*.

Referenciar a Silhouette de sklearn.

Silhouette És una mesura no supervisada, que valora la integrat de cada node dintre d'un cluster. Per cada punt (o observació) calculem la *silhouette* amb la següent fórmula:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

on:

i és el punt del qual volem calcular la *silhouette*.

$a(i)$ és la distància mitja als demés punts dintre del cluster de i.

$b(i)$ és la distància mitja als punts que no estan dintre del cluster de i.

Un cop tenim la *silhouette* calculada per cada observació, per tenir la *silhouette* del cluster, fem la mitja de totes elles.

$$\text{silhouette}_\text{g} = \frac{1}{n} \sum_{j=1}^{n} silhouette_i$$

4.2.2 Predicció

La etapa de predicció és important en un projecte de data science, ja que ens permet predir el futur d'una forma estadística en base a les observacions que tenim. Però igual que la clusterització, hi han diverses tècniques, aquí explicaré quines tècniques hem utilitzat per aquest projecte.

Un predictor és...

Explicar

lincar la teoria de TNUI - recommenders

Recomanador

Una de les tècniques per predir dades són els recomanadors. En aquest apartat explicaré com funciona el recomanador que he montat possant-nos en context del nostre projecte. Tenint en compte les notes d'un conjunt d'alumnes, el recomanador és capaç de predir de forma estadística les notes d'un alumne en base a la resta dels altres.

Imaginem que tenim una matriu tal que:

$$C = \begin{array}{c} \\ e_1 \\ e_2 \\ \vdots \\ e_n \end{array} \begin{array}{c} \begin{array}{cccc} a_1 & a_2 & \cdots & a_m \end{array} \\ \begin{pmatrix} c_{11} & ? & \cdots & c_{1m} \\ c_{21} & c_{22} & \cdots & ? \\ \vdots & \vdots & \ddots & \vdots \\ ? & c_{n2} & \cdots & c_{nm} \end{pmatrix} \end{array}$$

on:

- e_i és un estudiant.
- a_i és una assignatura.
- c_{ij} és la nota d'un estudiant i en una assignatura j.
- $?$ són notes no completes, perquè un alumne no ha cursat l'assignatura.

La finalitat del nostre recomanador, és omplir les notes que apareixen amb ? i possar la nota més adient. Abans d'explicar com funciona, introduiré els diferents tipus de recomanadors que podem tenir:

Recomanador col·laboratiu bassat en estudiant (RCxE) Prediem la nota d'un alumne en base a la semblança de l'alumne amb la resta. És a dir, si un alumne e_i té unes notes semblants a un alumne e_j, les assignatures que no ha cursat e_i podrem dir que seran semblants a les notes que ha tret e_j en aquelles assignatures.

Recomanador col·laboratiu bassat en assignatures (RCxA) Ara en comptes de bassar-nos en la semblança entre els estudiants, ens basem en la semblança entre una assignatura amb la resta. És a dir, si una assignatura a_i segueix una distribució semblant a una assignatura a_j, llavors podem dir que un alumne e_i treurà una nota semblant en ambdues assignatures.

Recomanador híbrid Per últim tenim la barreja dels dos recomanadors esmentats, aplicant un pes d'importancia a cadascun. Aquest recomanador no s'ha fet servir en aquest projecte, però es podria fer servir si es pogués apendre quin pes assignar a cada tipus de recomanador.

Començaré explicant el recomanador col·laboratiu bassat en l'estudiant, el qual agafaré com a base per explicar el bassat en assignatures. Imaginem que tenim una matriu semblant a la d'abans:

$$C = \begin{array}{c} \\ e_1 \\ \vdots \\ e_p \\ \vdots \\ e_n \end{array} \begin{array}{c} \begin{array}{ccccc} a_1 & \cdots & a_q & \cdots & a_m \end{array} \\ \begin{pmatrix} c_{11} & \cdots & \mathbf{c_{1q}} & \cdots & ? \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ ? & \cdots & \mathbf{?} & \cdots & c_{pm} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ c_{n1} & \cdots & \mathbf{c_{nq}} & \cdots & c_{nm} \end{pmatrix} \end{array}$$

El que volem és predir la nota que té el símbol ? en negreta a la posició c_{pq}. El que necessitem és aplicar a la posició que volem predir la següent fórmula:

$$c_{pq} = \sum_{i=1}^{n} \alpha_{e_p e_i} c_{iq}$$

on:

- α és una funció de similitud, que dóna pes a $c_{a_q e_j}$.
- e_i és un estudiant.
- a_i és una assignatura.

Amb aquesta fórmula podem veure la funcionalitat d'aquest recomanador, si ens fixem, com més semblants siguin dos estudiants, més pes li donarem a la nota que ha tret un dels dos per recomanar-li a l'altre. Aquesta fórmula és la fórmula d'una mitja ponderada.

Ara bé, si el que volem és fer un recomanador bassat en assignatures, tenim dues opcions. O bé aplicar la següent fórmula:

$$c_{pq} = \sum_{j=1}^{n} \alpha_{a_q a_j} c_{pj}$$

O bé, fer la transposada de la matriu anterior i aplicar la mateixa fórmula d'abans.

link de sklearn

Random Forest Regressor (RFR)

Abans d'explicar la tècnica de *Random Forest Regressor*, s'ha d'entendre el concepte d'un arbre de regressió. Un arbre de regressió és una tècnica utilitzada en aprenentatge automàtic, que es defineix com un model predictiu que mapeja observacions sobre una característica a conclusions sobre el valor objectiu d'aquesta característica. En aquestes estructures d'arbre, les fulles representen un valor real d'aquella característica i les branques les conjuccions de caracterítiques que han portat fins a la fulla.

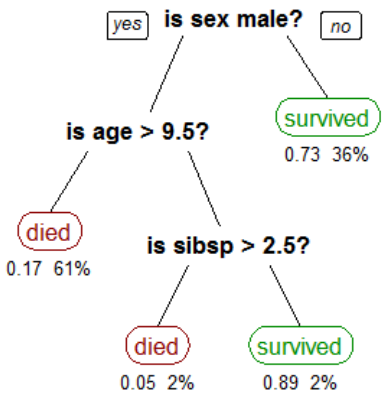


Figure 5: *CART tree titanic survivors* de Stephen Milborrow

Random Forest Regressor, és un conjunt d'arbres de regressió, on el resultat és la mitja de la sortida de cada arbre, a més per a cada arbre s'aplica un soroll aleatori a les dades sense variar en la seva distribució, això fa que es beneficiï al fer la mitja.

link de sklearn

Regressor lineal (LR)

Un regressor lineal modelitza una recta de regressió a partir d'un núvol de punts. La recta definida, és la recta més propera que passa per tots els punts. El que busca és definir una variable depenent a partir d'un conjunt de variables, és a dir:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon$$

on β_i són termes constants i n són els conjunts d'observacions que tenim. En el cas d'una sola variable depenent, tindriem un resultat com el de la figura següent:

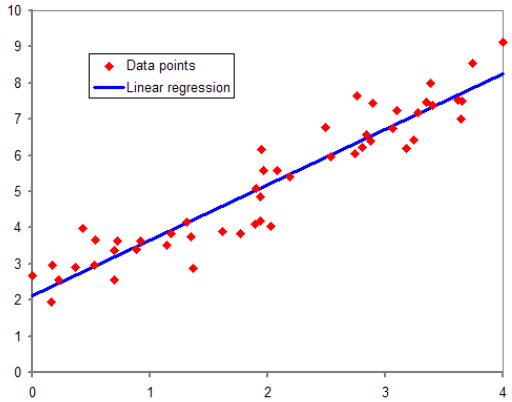


Figure 6: "Normdist regression" by Amatulic

link de sklearn

Mètriques utilitzades

Igual que en la secció de clusterització, per a la predicció de dades, també hem utilitzat mesures per validar les nostres prediccions. Aquí hem utilitzat mesures supervisades. Anomenem y_{pred} al conjunt de qualificacions que s'ha predit d'un alumne, i y_{test} al conjunt de qualificacions real del estudiant.

Error promig absolut (MAE) És una mesura supervisada que és basa en fer la mitja dels errors produïts pel predictor. Està definit per la següent fórmula:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |y_{pred_i} - y_{test_i}|$$

Error promig quadràtic (MSE) Per un altre banda tenim una segona mètrica supervisada també, però aquesta mètrica penalitza els error alts, ja que la diferència es elavada al quadrat, quedaria la següent fórmula:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_{pred_i} - y_{test_i})^2$$

Coeficient de pearson (PCC) El coeficient de pearson, l'utilitzem com una mètrica supervisada i la fem servir per mesurar la diferencia de la distribució de les notes predites amb les notes reals. El coeficient de pearson està definit per la següent fórmula:

$$\text{PCC} = \left| \frac{\sum_{i=1}^{n}(y_{pred_i} - \bar{y}_{pred_i})(y_{test_i} - \bar{y}_{test_i})}{\sqrt{\sum_{i=1}^{n}(y_{pred_i} - \bar{y}_{pred_i})^2 \sum_{i=1}^{n}(y_{test_i} - \bar{y}_{test_i})^2}} \right|$$

Desviació estàndard (std) També calculem la desviació estàndard per veure si els errors són més o menys dispersos. La fórmula utilitzada és la següent:

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(|y_{pred_i} - y_{test_i}| - \mu)^2}$$

Mean Rangking Score (MRS) Aquesta és una técnica desenvolupada per nosaltres, està basada en la mesura de *Error promig absolut*, però amb valors discrets. És una mesura supervisada per tal de mesurar quant de bo és un ranking que s'hagi predit. Es tracta d'una mitja a partir de les distancies d'error en el ranking. La métrica presenta la següent fórmula:

$$\text{MRS} = \frac{1}{n}\sum_{i=1}^{n}|\gamma(y_{pred_i}) - \gamma(y_{test_i})|$$

on:

γ és una funció que ens retorn la posició de l'elemtn en el ranking

La millor manera d'entendre aquesta mètrica és mostrar un exemple a partir d'aquesta taula:

Ranking Real	Ranking Predit
A1	A4
A2	A2
A3	A1
A4	A3

Table 3: Exemple de rankings

Per aquest exemple, hauriem de recorrer els 4 elements:

$$\left|\gamma(A1_{notes_r}) - \gamma(A1_{notes_p})\right| = |1-3| = 2$$

$$|\gamma(A2_{notes_r}) - \gamma(A2_{notes_P})| = |2-2| = 0$$

$$\left|\gamma(A3_{notes_r}) - \gamma(A3_{notes_p})\right| = |3-4| = 1$$

$$\left|\gamma(A4_{notes_r}) - \gamma(A4_{notes_pr})\right| = |4-1| = 3$$

Quedant:

$$\text{MRS} = \frac{2+0+1+3}{4} = \frac{6}{4} = 1.5$$

Ens podem fixar que si comparem dos rankings iguals, llavors MRS = 0. Per tant, com més proper estigui a 0, millor s'aproparà la predicció del ranking real.

Totes aquestes mètriques són necessaries per evaluar cada tècnica de predicció que utilitzo. Tot i així, les tècniques més importants i que tenen més pes són l'error promig absolut i quadràtic.

4.2.3 Reducció de dimensions

Una de les últimes tècniques que utilitzo en aquest projecte d'innovació docent és la reducció de dimensions. És imprescindible per poder visualitzar les teves dades si tenen una dimensió major que 3. Aquestes tècniques a més permeten reduïr el cost computacional sense variar en el seu resultat. Una de les tècniques utilitzades en aquest projecte és l'anàlisis de components principals (PCA).

link de sklearn

PCA

L'anàlisis de components principals o PCA el que fa és escollir un nou sistema de coordenades a partir d'una transformació lineal on s'ordenen les variances per mida. La variança amb major mida s'escollirà com eix principal, la segona variança com a segon eix, així successivament fins obtenir la dimensionalitat escollida per argument.

5 Experiments i resultats

En aquest apartat s'explicarà pas per pas els resultats obtinguts per cada pregunta plantejada. Fins ara s'han llegit tots els conceptes necessaris per poder entendre aquesta secció de la documentació. Començaré amb les preguntes relacionades amb la clusterització i acabaré amb els resultats obtinguts amb la predicció de notes.

Abans de començar a comentar els resultats, explicaré de quines dades parteixo per respondre cada pregunta. D'inicial tenim tota una taula on cada fila és la qualificació d'un alumne donada un assignatura, per tant en cada fila tenim informació com *l'identificador d'alumne, assignatura, tipus d'apunt (convalidada, ordinaria o de reconeixement), qualificació de l'assignatura, ...* És a partir d'aquesta taula que fem una conversió de tal manera que en cada fila ens quedi un alumne i cada columna sigui una assignatura, construint una matriu tal que així:

$$\begin{array}{c} C \\ e_1 \\ e_2 \\ \vdots \\ e_n \end{array} \begin{array}{c} \begin{array}{cccc} a_1 & a_2 & \cdots & a_m \end{array} \\ \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1m} \\ c_{21} & c_{22} & \cdots & c_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nm} \end{pmatrix} \end{array}$$

on:

e_i és un estudiant.

a_i és una assignatura.

c_{ij} és la nota d'un alumne donada una assignatura.

C és una matriu amb coeficients reals, $C \in M_{nxn}(\mathbb{R})$ on $0 \leq c_{ij} \leq 10$, és a dir aquesta matriu no conté cap nombre desconegut i que cada alumne e_i ha cursat tot el conjunt d'assignatures $\{a_1, a_2, a_3, \ldots, a_m\}$.

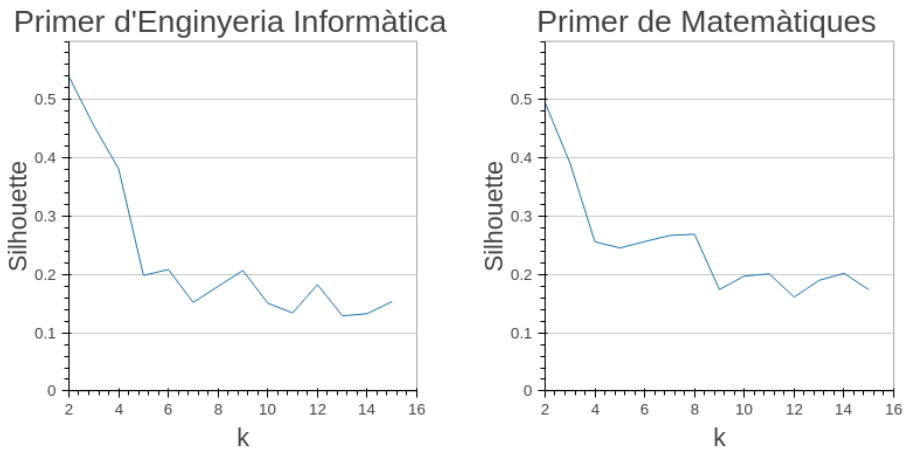
El conjunt d'assignatures que apareixen en les columnes pot variar depenent de la pregunta que volem respondre, pot ser el conjunt d'assignatures de primer, com el conjunt de les de primer més les de segon. Però a partir d'una matriu C com aquesta em basaré algunes qüestions.

5.1 Hi ha diferents perfils d'alumnes?

La resposta a aquesta pregunta és trobar diferents tipus d'estudiants en relació a la seva nota (alumnes amb notes molt bones en tot, alumnes amb males notes en certes assignatures, alumnes que suspenen, entre altres). Però volem que el nostre algoritme explori els grups que hi han.

Com el que busquem són alumnes amb qualificacions semblants, utilitzarem la tècnica de *K-means*, ja que pot agrupar alumnes en relació a la distància de les seves notes. Però clar, *K-means* té una limitació, necessita com argument el número de clusters que volem segmentar. Hem de trobar una forma de poder trobar la millor k.

La primera opció que vam pensar és aplicar *K-means* amb diferents k i per cada prova, calcular la mesura de *Silhouette*. L'algoritme de *K-means* rep com a paràmetre una matriu com la matriu C amb els alumnes que hagin cursat totes les assignatures de primer de cada grau implantat en la Facultat de Matemàtiques de la Universitat de Barcelona.



(a) Informàtica (b) Matemàtiques

Figure 7: Càlcul de la mesura *Silhouette*

Aquest gràfic ens diu que la millor k en ambdós casos és $k = 2$ i la mesura de silhouette descendeix conforme augmenta el paràmetre k. Però clar aquest resultat no ens interesa, perquè busquem un número de clusters major que 2, encara que els clusters estiguin menys disgregats. Per tant com aquesta tècnica no ens serveix hem de buscar una altre forma per determinar quina és la millor k.

L'altre solució proposada és reduïr la dimensionalitat de les dades per tal de poder visualitzar-les en un pla dos-dimensional. Per poder fer això podem aplicar la tècnica de PCA per reduïr de 10 dimensions a 2. Un cop visualitzem cada estudiant en un espai 2D, podem aplicar un algoritme d'agrupació com *Mean Shift* per veure quants clusters hi han i poder determinar una k per cada curs.

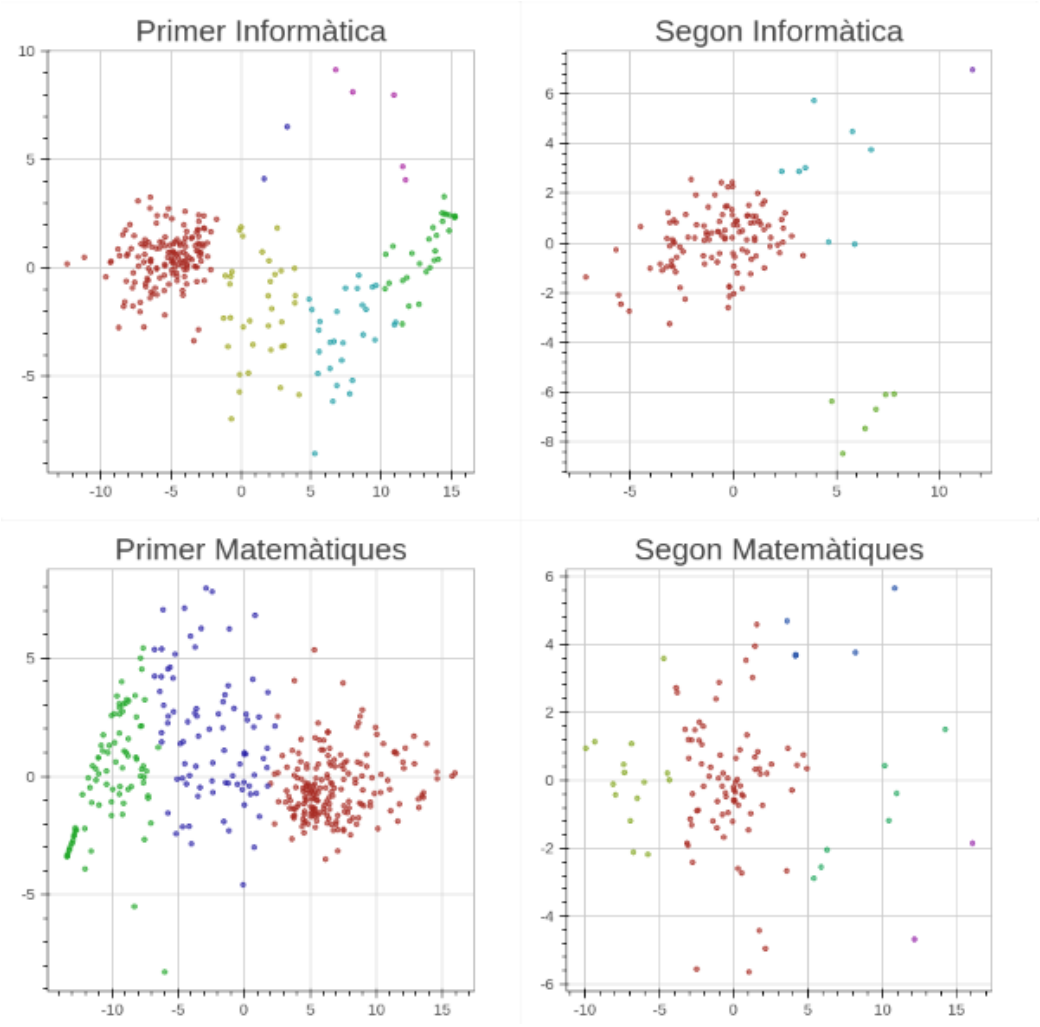


Figure 8: Mean Shift després d'aplicar PCA

Ara es poden distingir millor el número de clusters o agrupacions que trobem per cada curs.

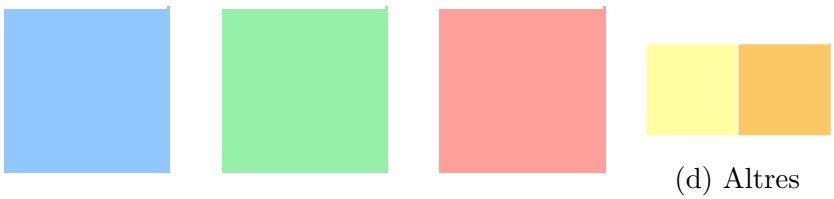
Primer d'enginyeria Informàtica Ens separa tot el conjunt de punts en 6 agrupacions (*vermell, beix, blau claret, verd, lila i blau fosc*), però el grup *blau fosc i lila* són un grup tan reduït i separat de la resta que el podriem comptar com un sol cluster. $k = 5$

Segon d'Enginyeria Informàtica Per a aquest curs ens separa als estudiants en 4 grups, i podem veure que els clusters estan força disgregats entre ells i no fa falta unificar cap. $k = 4$

Primer de Matemàtiques Per a primer del grau de Matemàtiques ens separa les observacions en 3 clusters. Com no es veu cap anomalia, a part de la petita separació dels petits punts verds, podem considerar els tres clusters. $k = 3$

Segon de Matemàtiques Aquest és el curs que m'ha donat més problemes, perquè té els punts més distanciats entre ells i això fa que no es pogui interpretar el número de clusters per aplicar *K-means*. Més endavant veurem que el número de clusters òptim és 3, ja que amb 4 ens dóna dos clusters molt semblants, els quals es poden unificar. $k = 3$

Ara que ja tenim el valor de k adeqüat per cada curs, podem aplicar la tècnica de *K-means*. S'ha utilitzat una combinació de colors adequada per cada perfil.



(a) Aprovats (b) Bones notes (c) Suspesos (d) Altres

Figure 9: Categoria de colors utilitzada per representar els perfils d'estudiants

Començarem comentant els resultats obtinguts amb el curs de primer d'Enginyeria Informàtica on hem dit que aplicariem *K-means* amb $k = 5$.

retocar la imatge, per treure el quadrat gris

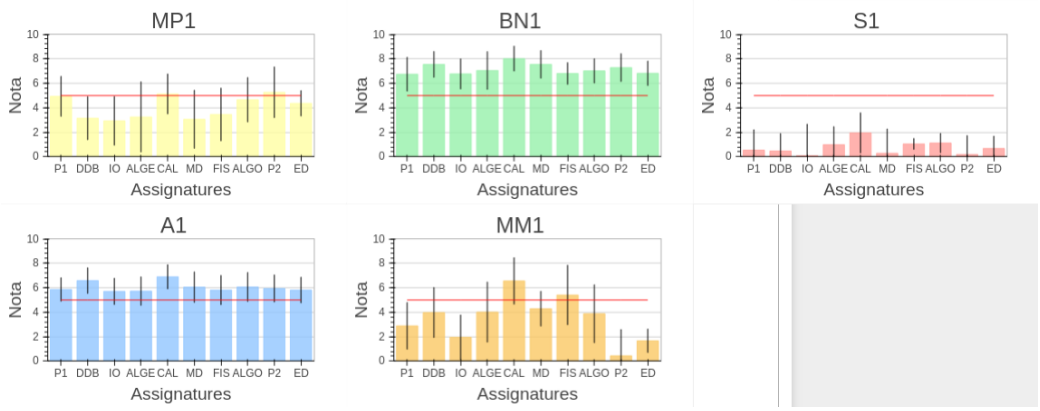


Figure 10: Perfils d'alumnes de primer d'Enginyeria Informàtica

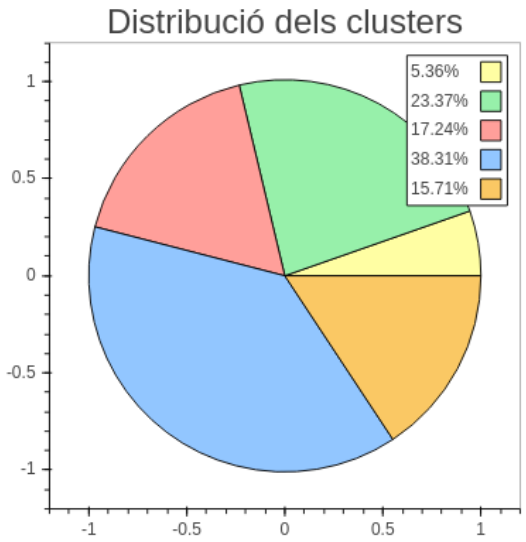


Figure 11: Percentatges de cada agrupació

Primer explicaré que és el que veiem en aquests gràfics i així poder interpretar els demés. Cada gràfic són els diferents perfils d'estudiants que ha trobat la tècnica de *K-means*. En cadascun d'ells trobem com a títol l'etiqueta assignada a aquell perfil i el color de cadascún depén de la categoria explicada abans. En l'eix d'abscisses es veuen les assignatures (en aquest cas les de

primer d'Enginyeria Informàtica), i en l'eix d'ordenades tenim la mitja de cada assignatura (longitud de la barra). Per últim les línees negres determinen la desviació estàndard de la distribució de cada assignatura i la línea vermella és una marca per identificar l'alçada de l'aprovat.

MP1 - Millors en programació de primer d'Informàtica Aquest perfil encaixa amb els alumnes que tenen millores notes en les assignatures de programació que en les de matemàtiques. La distribució d'aquest és molt dispersa, això ho podem veure per la llargada de la barra negra (desviació estàndard). També és, perquè la mostra és petita, representa el 5.36% de la mostra total.

BN1 - Bones notes de primer d'Informàtica Aquest perfil correspon als alumnes que tenen bones notes en totes les assignatures de primer i com podem veure en el gràfic de pastilla, representen un 23.37% del total, sent el segon perfil més abundant. Podem veure ara que la mostra és més gran, la desviació estàndard és menor, és a dir, aquest perfil és força estable, tots els estudiants que hi pertanyent es distancien amb una qualificació promig d'1.5 aproximadament.

S1 - Suspesos de primer d'Informàtica Com podem veure, aquest perfil són els estudiants que suspenen la majoria d'assignatures. A primeres podem pensar que són els que solen deixar la carrera i és això el que anem a respondre en la següent pregunta plantejada. També podem veure que són un 17.2 % del total d'alumnes que han cursat les assignatures de primer, no són un percentatge baix.

A1 - Aprovats de primer d'Informàtica Són la major part dels alumnes, amb un 38.31% del total, i són els alumnes que de mitja treuen entre 5 i 7. Igual que passa amb el cluster *BN1*, la desviació estàndard de cada assignatura és força baixa, i això fa que el cluster sigui consistent.

MM1 - Millors en matemàtiques de primer d'Informàtica Aquest perfil igual que *MP1*, és bastant inestable, ja que tenen una desviació estàndard alta, és a dir, hi ha una diversitat de notes elavada, no es concentren tots els alumnes a tenir la mateixa nota. Tot i que siguin dispersos, són un 15.71 % del total, un percentatge forç alt.

Fent un anàlisis general de les gràfiques, podem veure que en l'assignatura *Estructura de Dades* (ED) presenta sempre una desviació més petita que la

resta d'assignatures en cada perfil, és a dir, que les notes en concentren més en la mitja marcada. Per un altre banda també es pot veure que en tots els perfil, sense mirar la qualificació corresponent, la nota més alta en els 6 perfils és de l'assignatura de *Càlcul* (CAL).

Seguint amb l'anàlisis podem veure també els perfils de tots els alumnes que hagin cursat totes les assignatures de segon, on he aplicat *K-means* amb $k = 4$.

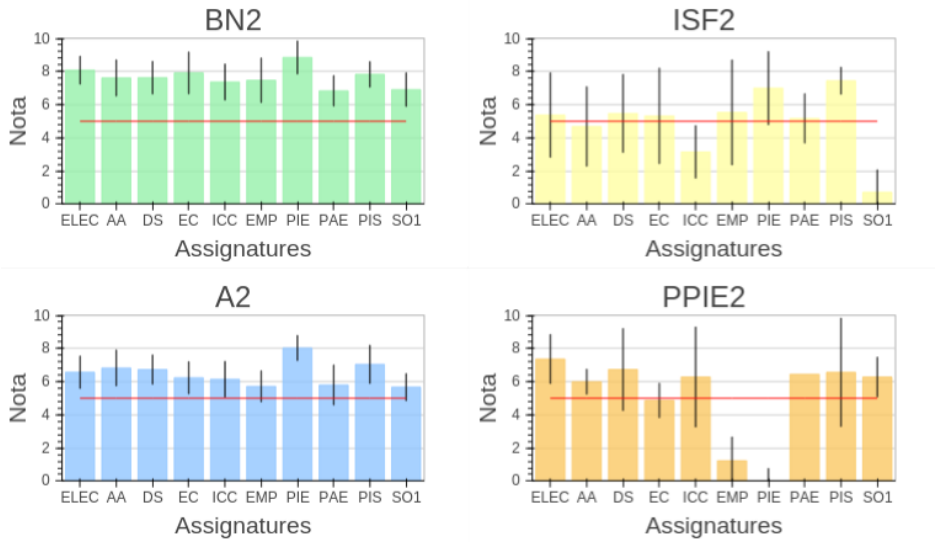


Figure 12: Perfils d'alumnes de segon d'Enginyeria Informàtica

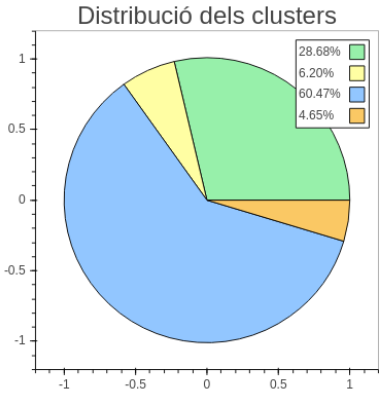


Figure 13: Percentatges de cada agrupació

BN2 - Bones notes de segon d'Informàtica Aquest és un perfil força semblant a *BN1*, és per això que ens plantejem la pregunta: *Cadascun d'aquests perfils amb quin perfil de provinença encaixa?*, és a dir, és cert que els que tenen bones notes a primer, solen ser els que treuen bones notes a segon, el que anomenem conservació de clusters. De tots els alumnes que han cursat segon un 28.68% pertanyen a aquest cluster, més d'un quart de la mostra.

ISF2 - ICC i SO1 fluixes Aquest perfil encara que sigui minoritari, amb un 6.2% del total, és força curiós, ja que són alumnes que tenen *Introducció a la Computació Científica* (ICC) i *Sistemes operatius I* (SO1) amb notes més baixes que la resta. També s'ha de dir que tot i que les mitjes siguin més petites, les desviacions estàndards són molt altes, el que fa que les notes dels alumnes siguin més diverses i no segueixin exactament la distribució de mitjes de cada perfil.

A2 - Aprovats de segon d'Informàtica Aquest perfil és semblant al perfil *A1*, pertany als alumnes que tenen notes entre 5 i 7. Són el 60.47% del total d'alumnes que han cursat les assignatures de segon.

PPIE2 - Problemes amb PIE i Empresa Aquest és el perfil amb un percentatge més petit de població, un 4.65%. És un perfil força dispers, ja que les desviacions estàndar són altes, i això es pot veure en assignatures com PIS o ICC. A més és curiós perquè és un grup que apareixen amb *Empresa* (EMP) i *Probabilitat i estadística* (PIE) suspeses, però no he trobat un perquè a aquest fenòmen.

Ens podem fixar que la distribució de mitjes del perfil *BN2* i *A2*, és força semblant, només que *BN2* té les mitjes més altes.

Deixant enrere al grau d'Enginyeria Informàtica, em poso a analitzar als estudiants del grau en matemàtiques. Comencem amb els alumnes de primer, els quals els segmentem en 3 agrupacions ($k = 3$).

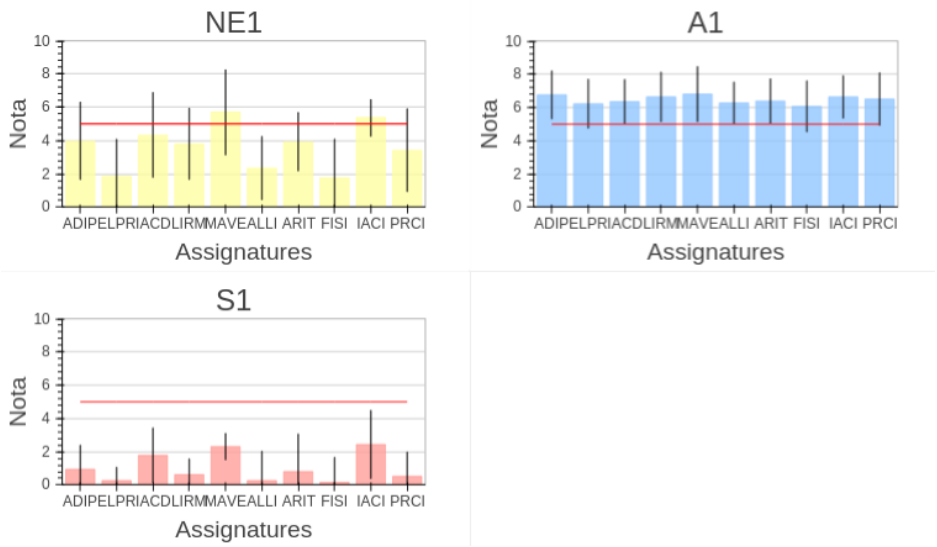


Figure 14: Perfils d'alumnes de primer de Matemàtiques

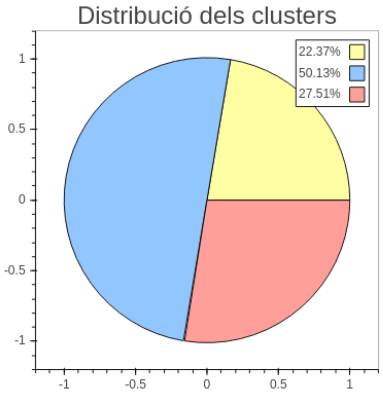


Figure 15: Percentatges de cada agrupació

NE1 - No estables de primer de Matemàtiques Podem veure que aquest cluster té les mostres molt distanciades, per les desviacions estàndards que presenta. Aquest perfil l'he clasificat com *No estables de primer*, ja que són alumnes que amb prou feines poden aprovar certes assignatures. Conformen un 22.37% del total d'alumnes.

A1 - Aprovats de primer de Matemàtiques Aquest perfil pertany als estudiants que tenen totes les assignatures aprovades de mitja i són els que formen la major part del total, amb un 50.13%.

S1 - Suspesos de primer de Matemàtiques Per últim, i sense faltar, tenim els alumnes que de mitja suspenen totes les assignatures. Conformen un 27.51% del total d'estudiants.

Els resultats són força esperats per aquest curs, tenim els que ho aproven tot i els que ho suspenen tot, i per un altre banda tenim la resta que no pertanyen ni a un ni a l'altre.

Per últim tenim als alumnes de segon de Matemàtiques, on no s'observa res interessant com tenim als perfils de Informàtica.

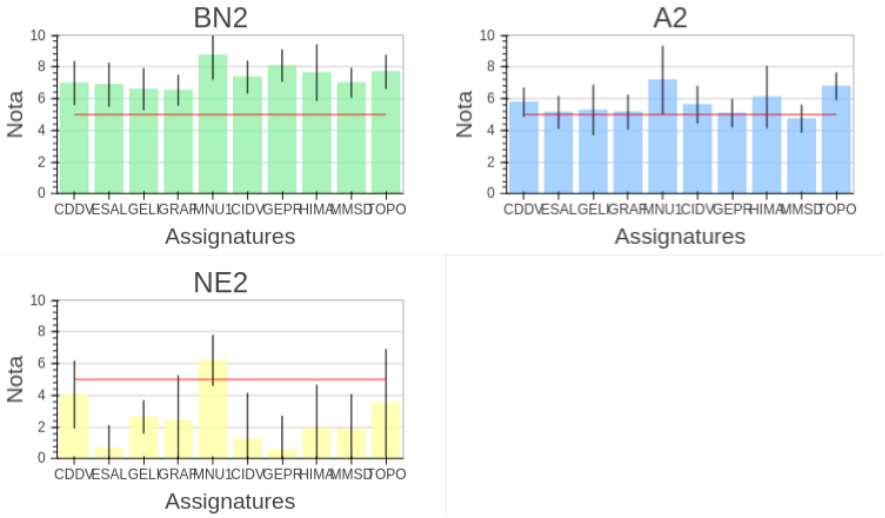


Figure 16: Perfils d'alumnes de segon de Matemàtiques

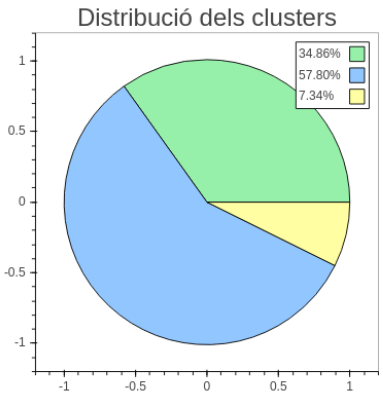


Figure 17: Percentatges de cada agrupació

BN2 - Bones notes de segon de Matemàtiques Torna a aparèixer aquest tipus de perfils, alumnes amb mitjes de qualificacions altes. Tot i així, és sorprenent el percentatge d'alumnes que pertanyen a aquest cluster, un 34.86%.

A2 - Aprovats de segon de Matemàtiques Per un altre banda tornem a tenir els alumnes amb qualificacions en el rang d'aprovat, tot i que la majoria freguen la línea de l'aprovat, com és en tots els casos, aquest és el perfil més abundant, amb un 57.80%.

NE2 - No estables de segon de Matemàtiques Igual que a primer del grau de Matemàtiques tenim el perfil de *No estables*, aquí el tornem a tenir, tot i que aquest perfil només té aprovada per mitja una assignatura, *Mètodes numèrics I* (MNU1). Pertanyen al 7.34% del total.

En aquest curs tenim un efecte semblant a primer d'Enginyeria Informàtica amb l'assignatura de *Càlcul*, hi ha una assignatura que en tots els perfils té la mitja més alta, *Mètodes numèrics I* (MNU1).

5.2 Quina es la tassa d'abandonament per cada tipus de perfil?

És cert que els que suspenen abandonen la carrera? Ara ho podrem demostrar amb dades. Ens centrarem en la tassa d'abandonament dels alumnes que cursen primer, tant d'Enginyeria Informàtica com el grau de Matemàtiques. Tornaré a mostrar els perfils per poder contrastar cada perfil amb la seva tassa d'abandonament.

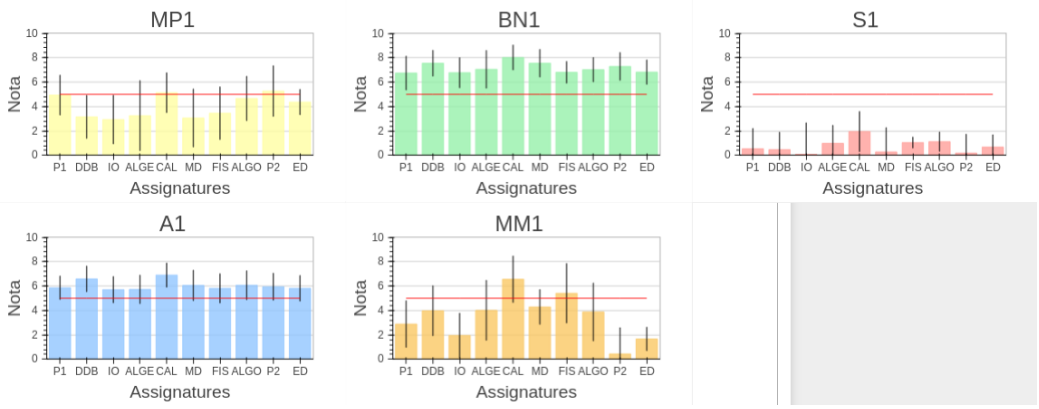


Figure 18: Perfils d'alumnes de primer d'Enginyeria Informàtica

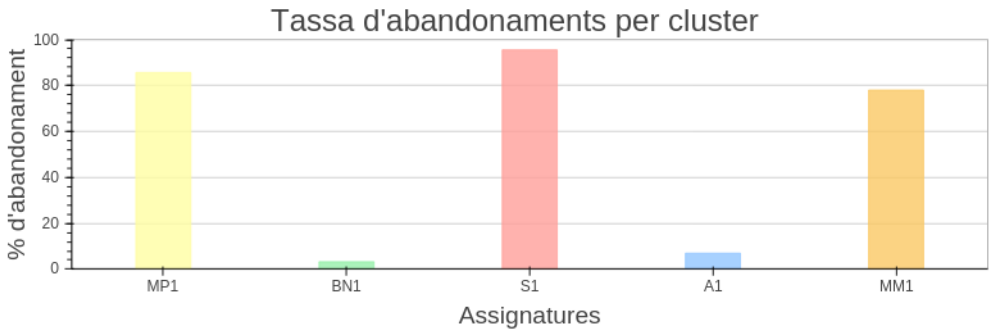


Figure 19: Tassa d'abandonaments per perfil

Com era d'esperar, els alumnes que amb més probabilitat deixen la carrera són els que suspenen, seguits d'estudiants que no tenen unes notes massa estables. És a dir, que la majoria que passen a segon pertanyen al cluster *BN1* i *A1*. Però si ens fixem una mica en el gràfic d'abandonaments, els perfils d'aprovats i de bones notes, hi ha un petit percentatge indicat que diu que abandonen la carrera. Bé, en les dades que tenim, no tenim un camp que

ens indiqui si un alumne ha abandonat la carrera o no, ja que no ho tenen registrat, suposo perquè potser pot tornar un altre any l'estudiant. Com ho hem fet llavors? Hem agafat per cada perfil tot el conjunt d'alumnes d'aquell perfil i s'ha comprovat per cadascún d'ell si té assignatures matriculades a l'any següent. Però clar, hi han alumnes del darrer any que s'han de matricular per l'any vinent encara (i no apareixen matriculats a l'any següent), per això hi ha un petit marge d'error i els clusters *BN1* i *A1* aparèixen amb una mínima tassa d'abandonaments.

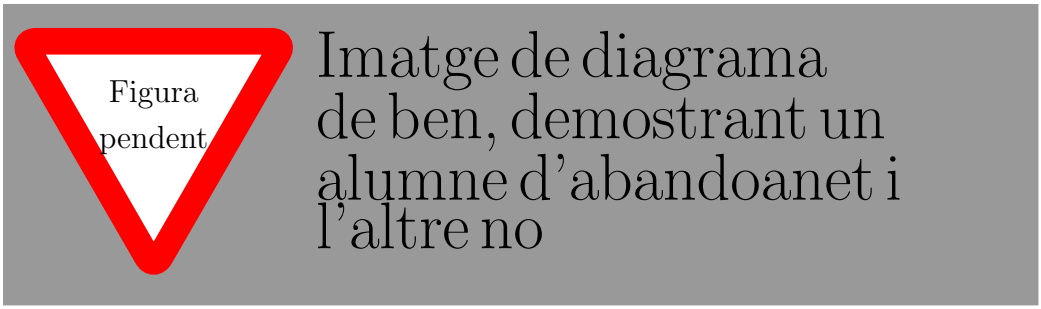


Figure 20: Some caption

Ara, per altre banda tenim el curs de primer de Matemàtiques, on apliquem els mateix algoritme explicat en el paràgraf anterior. També trovem un petit marge d'error en els perfils que ho aproven tot.
Igual que en Enginyeria Informàtica, ens trobem que la major tassa d'abandonament es troba en els alumnes que suspenen per mitja totes les assignatures. Seguidament els hi segueixen els estudiants que no tenen unes notes massa regulars i com s'ha comentat anteriorment, els alumnes que estan classificats com *Aprovats*, també surten amb una petita tassa d'abandonament.

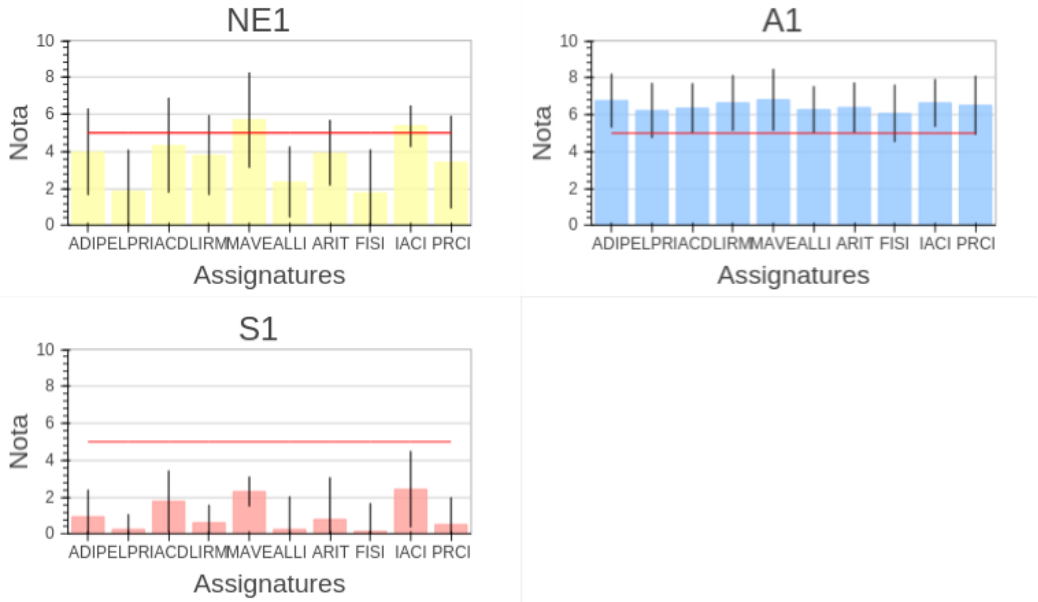


Figure 21: Perfils d'alumnes de primer d'Enginyeria Informàtica

5.3 Cadascun d'aquests perfils amb quin perfil de procedència encaixa?

El que es vol mirar en aquesta pregunta és la conservació de clusters, per exemple, els estudiants que treuen bones notes a primer, segueixen treien bones notes a segon? O, de quina via d'accés solen provenir els estudiants de primer de matemàtiques?

Començarem, com hem fet anteriorment, amb els estudiants que han cursat primer d'Enginyeria Informàtica. Com s'ha explicat en la secció de preguntes plantejades, contrastem primer d'Enginyeria Informàtica amb les vies d'accés: Batxillerat i salt d'Universitat.

En l'interior del cercle, podem veure el mateix gràfic de pastilla que s'ha vist anteriorment dels perfils dels estudiants de primer d'Enginyeria Informàtica (AQUÍ). Per sobre de cada perfil es veu en quantitat d'on venen els alumnes del perfil.

Es pot veure com en tots els perfils, venen meitat de Batxillerat i meitat de Salt d'Universitat aproximadament. Es pot distingir que els estudiants classificats com *Suspesos* solen venir més de Batxillerat que no pas d'una altre Universitat, tot i que la diferència és petita. Com s'ha vist en la gràfica

Canviar els colors de la gràfica

Possar referència a la figura anterior

Possar referen-

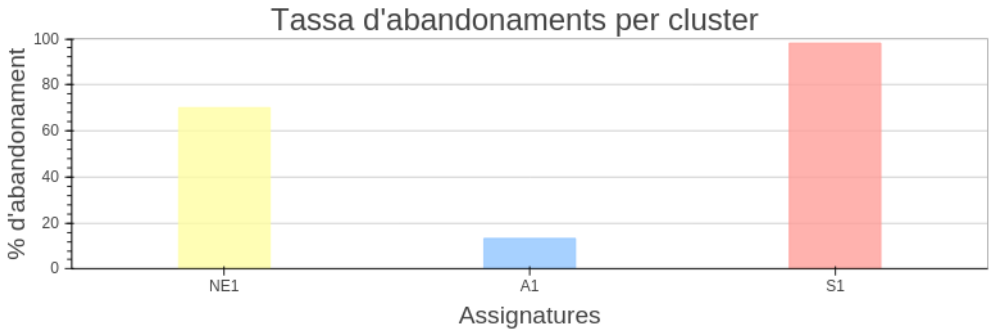


Figure 22: Tassa d'abandonaments per perfil

d'abandonament (AQUÍ), els alumnes que passen amb més abundancia a segon són els classificats com *A1* i *BN1*, per tant procedim a eliminar a la resta, ja que són minoria, i així més llegible la següent gràfica.
Ara és veu amb més claredat el significat d'un gràfic com aquest, ja que podem veure com els que solen treure bones notes a primer dEnginyeria Informàtica, solen treure bones notes a segon també, i els que classificaven com *Aprovats de primer*, solen parar a *Aprovats de segon*. També podem veure com un petit percentatge d'alumnes que treuen notes a primer, passen a treure notes més baixes a segon, igual que alumnes etiquetats com *Aprovats de primer* amb una petit quanitat paren a perfils inestables com *ISF2* o *PPIE2*.

Figure 23: Conservació de clusters dels alumnes de primer d'Informàtica

Per últim es veurà la provinença dels estudiants de primer del grau de Matemàtiques, on també es pot veure una tendencia.
En aquesta gràfica es veu amb claredat que la major part d'estudiants que han cursat primer del grau de Matemàtiques, venen de Batxillerat.
Una de les raons per les quals vam plantejar aquesta pregunta, era per saber si podiem determinar un perfil d'estudiant al grau a partir de la via d'accés d'aquest. Malauradament no s'ha trobat cap correlació com aquesta, però s'han pogut veure resultats molt coherents i que no esperavem trobar.

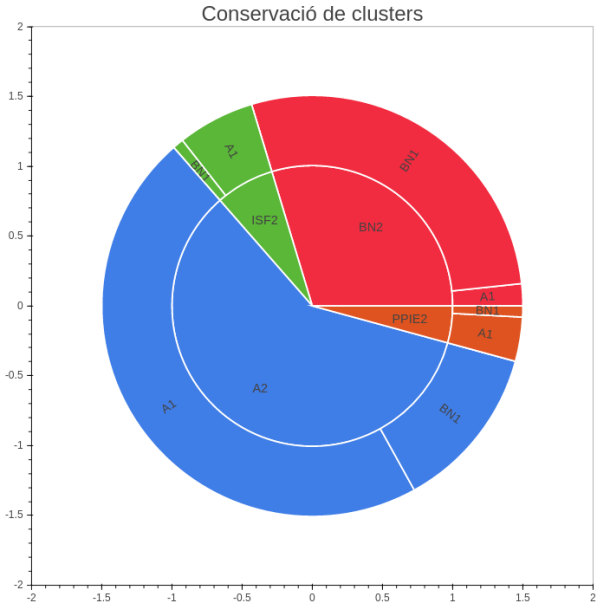


Figure 24: Conservació de clusters dels alumnes de primer d'Informàtica

5.4 Predicció de notes i ranking de dificultat d'assignatures

Per últim en aquest apartat s'explicará les tècniques de predicció que hem fet servir i quins resultats, tant quantitatius com qualitatius, s'han obtingut. La resposta que busquem d'aquesta pregunta, és poder mostrar un ranking, personalitzat per cada alumne, d'assignatures ordenades per la dificultat (per notes) que li pot arribar a costar a cadascún.

Abans de construïr el ranking, necessitem saber quina és la tècnica de predicció que s'ajusta millor a les nostres dades. Les técniques utilitzades per aquest experiment han sigut les següents, ja anteriorment explicades:

- Recomanador col·laboratiu bassat en estudiant
- Recomanador col·laboratiu bassat en assignatures
- *Random Forest Regressor*
- Regressor lineal

El primer pas és la construcció d'un recomanador col·laboratiu, com l'explicat a la secció de Recomanadors, seguint l'interfaç dels predictors de la biblioteca informàtica de *Scikit-learn*. Un predictor en *sklearn* ha de tenir un mètode

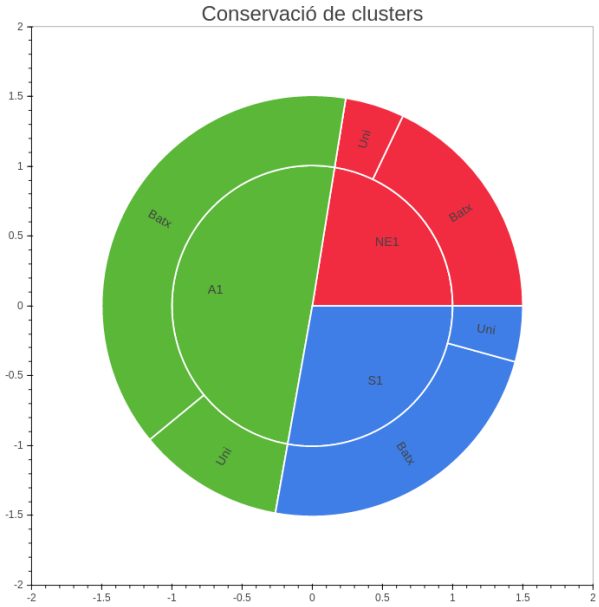


Figure 25: Conservació de clusters dels alumnes de primer d'Informàtica

fit i un altre que sigui *predict*. En la següent figura es pot veure un petit trocet de codi de com es fan les crides a les funcions.

```
from sklearn.linear_model import LinearRegression
# Definim les variables
X = [(1,2,3), (5,3,2), (3,1,0), (123,433,452), (233,231,786)]
# Fem el promig per cada Xi
y = [2, 3.33, 1.33, 336, 416]
# Definim una nova variable
x = (45,34,65) # promig = 48
# Creem un predictor
predictor = LinearRegression()
predictor.fit(X,y)
print predictor.predict(x)
# >> 47.9271056831
```

Figure 26: Fluxe de crides de funcions d'un predictor d'sklearn

En la figura de sobre és pot veure el flux de les crides a les funcions d'un predictor d'sklearn. Primer s'ha de cridar al mètode *fit* i seguidament ja podem cridar al mètode *predict* tantes vegades com volguem per predir.

```
class Recomender(BaseEstimator):
    def __init__(self, method=coefPearson, transpose=False):
        self._m = None
        self._method = method
        self._transpose = transpose

    def fit(self, X, y):
        # ...
        return self

    def predict(self, X):
        # ...
        return predicted
```

Figure 27: Estructura del recomanador col·laboratiu

Aquesta és la estructura del recomanador que s'ha construït. El constructor accepta per paràmetre la funció de similitud entre items (per defecte el coeficient de pearson), i un boleà que determina si és un Recomanador col·laboratiu basat en alumnes (*transpose=False*) o un Recomanador col·laboratiu basat en assignatures (*transpose=True*).

Un cop ja tenim llest tots els predictors, anem a mesurar quantitativament quin dels 4 predictors s'adequa més a les nostres dades. Les mètriques que s'utilitzaran són les que ja s'han explicat en la secció de *Mètriques de predictors*.

Proves quantitatives

Totes les proves realitzades s'han fet a partir de les notes d'Enginyeria Informàtica. Les proves es basen en dividir les dades en *training* i *test*. *Training* és el conjunt de dades que fem servir per entrenar el predictor (a través de la funció *fit*). *Test* és el conjunt de dades que utilitzem per predir (funció *predict*) i així comprovar si el predictor és lo suficient bo o no. Les proves s'han realitzat a partir de dos conjunts diferents de dades:

- Una on s'aplica com a *training* les notes de primer d'Enginyeria Informàtica i com a *test* les notes de segon.
- Per un altre banda apliquem un *training* amb les notes de primer i segon, i un *test* amb les notes de tercer d'Enginyeria Informàtica.

Per cada una de les proves s'han realitzat diferents proves, que són les següents:

Proves amb dades continues Provem els predictors mirant l'error produït amb les notes realsi les predites qualificades del 0 al 10. Les mesures utilitzades en aquesta prova són:

- **Error Promig Absolut** Per calcular la diferència mitja d'error produït.
- **Error Promig Quadràtic** Per veure si els errors són molt elevats.
- **Coeficient de Pearson** Per veure si la distribució entre les notes es manté.
- **Desviació estàndard** Utilitzada per veure si els errors es concentren o estan molt disgregats.

A més a partir dels errors produïts per cada predictor, es mostra un diagrama de caixes per representar la distribució dels errors. Més endavant amb un dels resultats s'explicarà l'interpretació del diagrama de caixes. Utilitzem un 90% de *training* i un 10% de *test*.

Proves amb dades discretes Igual que comprovem l'error produïr amb valors continus, del 0 al 10, ara etiquetem les notes segons el seu rang. El rang que s'ha utilitzat és el següent:

- **Suspés** Notes inferiors a 5 ($nota < 5$)
- **Aprovat** Notes entre 5 i 7 ($5 \leq nota < 7$)
- **Notable** Notes entre 7 i 9 ($7 \leq nota < 9$)
- **Excel·lent** Notes superiors a 9 ($nota \geq 9$)

Per poder visualitzar aquesta prova s'ha utilitzat una matriu de convulsió, on es representa amb un mapa de color. En la següent secció s'explicarà que és un mapa de color juntament amb els resultats obtinguts. Utilitzem un 50% de *training* i un 50% de *test*.

Proves amb ranking d'assignatures Per últim tenim les proves realitzades per mesurar el ranking que volem desenvolupar. La mètrica utilitzada ha sigut la *Mean Ranking Score*, que com ja s'ha explicat anteriorment, és una mètrica que ens permet mesurar quant de bó és un ranking.

Resultats de les proves quantitatives

Començarem amb les proves amb dades continues amb els dos conjunt de dades que s'han explicat en l'apartat anterior. Utilitzem un 90% de *training* i un 10% de *test*.
En aquesta figura podem veure un diagrama de caixes, aquest tipus de diagrama representa visualment una distribució. Cada caixa té les mateixes característiques: la línea del centre representa la mediana de la distribució, la caixa es visualitza un 50% de la mostra (des del primer quartil, fins al 3), les línes verticals determinen el límit de les distribucions i per últim tota mostra que etigui fóra de lo normal (mostra atípica) és visualitza per fóra dels límits amb un punt.

La distribució que podem veure a la figura són els errors que es presenten en cada predicció, és a dir, un predictor perfecte, mostrarà una distribució uniforme en el 0. En la figura es pot veure els quatre predictors utilitzats i quina distribució d'error segueix cadascún. Veiem que el predictor més estable i que concentra millor els errors són el recomanador col·laboratiu basat en estudiant i el *random forest regressor*. Per un altre banda tenim el Regressor linear que presenta la mediana més baixa, tot i que el tercer

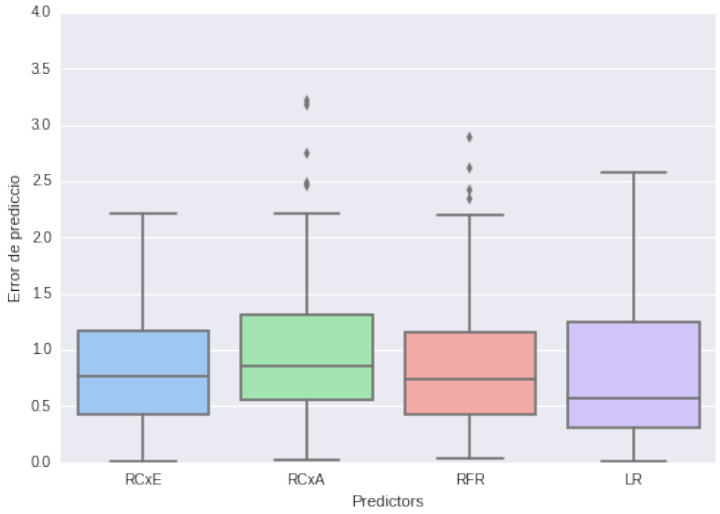


Figure 28: Prova amb dades continues. Traning: primer

quartil és massa alt. Els *outlayers* (mostres atípiques) que veiem tenen una explicació. Què passa si un alumne té un imprevist i no pot cursar una assignatura ja matrículada? Aquest factor no el contemplen els predictors. És per això que tenim observacións atípiques, perquè els predictors potser prediuen que un alumne treurà un 7, però l'alumne per qualsevol raó es desmatricula de l'assignatura, llavors en aquella assignatura li queda un 0. Més avall es mostra un exemple d'una mostra atípica amb una prova qualitativa.
Les mateixes conclusions podem extreure de les mètriques que utilitzem per fer aquesta prova:

Algoritme\Mètriques	**MAE**	**MSE**	**PCC**	**std**
RCxE	0.796	0.910	0.578	0.526
RCxA	1.006	1.509	0.309	0.705
RFR	0.868	1.152	0.505	0.632
LR	0.809	1.071	0.575	0.646

Table 4: Mètriques per a proves quantitatives amb dades continues. Traning: primer

Observem amb les mètriques que els millors predictors són RCxE, RFR i LR, tot i el que presenta menys errors és RCxE ja que té el l'error promig quadràtic més baix.
Igual que hem fet proves amb les notes de primer com a *training* i les de segon com a *test*, ara visualitzarem els mateixos resultats, però com a *training*

tenim les notes de primer juntament amb les de segon i com a *test* les notes de tercer.

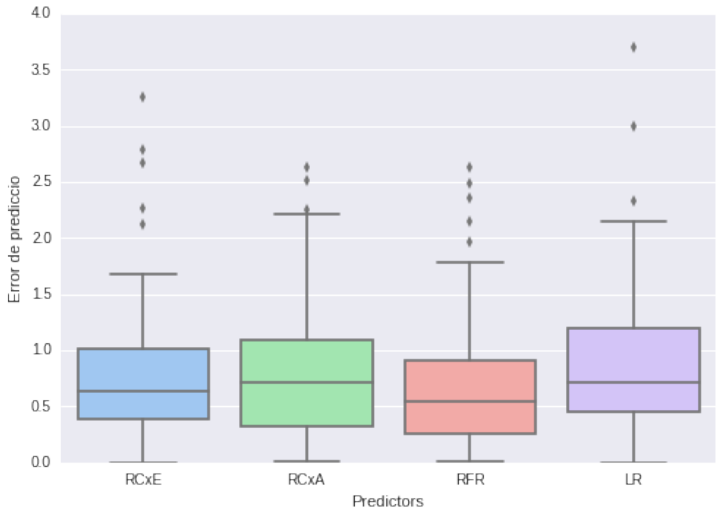


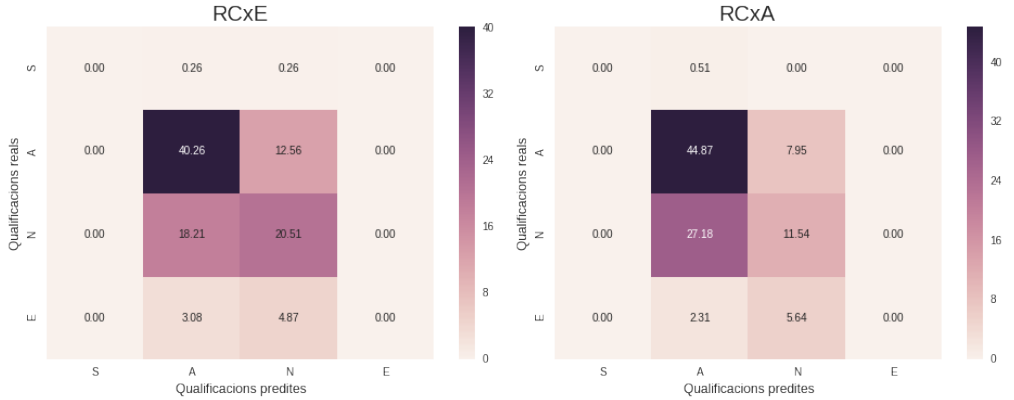
Figure 29: Prova amb dades continues. Traning: primer i segon

Algoritme\Mètriques	**MAE**	**MSE**	**PCC**	**std**
RCxE	0.869	1.590	0.276	0.914
RCxA	0.906	1.781	0.141	0.980
RFR	0.786	1.522	0.339	0.951
LR	0.932	1.890	0.283	1.011

Table 5: Mètriques per a proves quantitatives amb dades continues. Traning: primer i segon

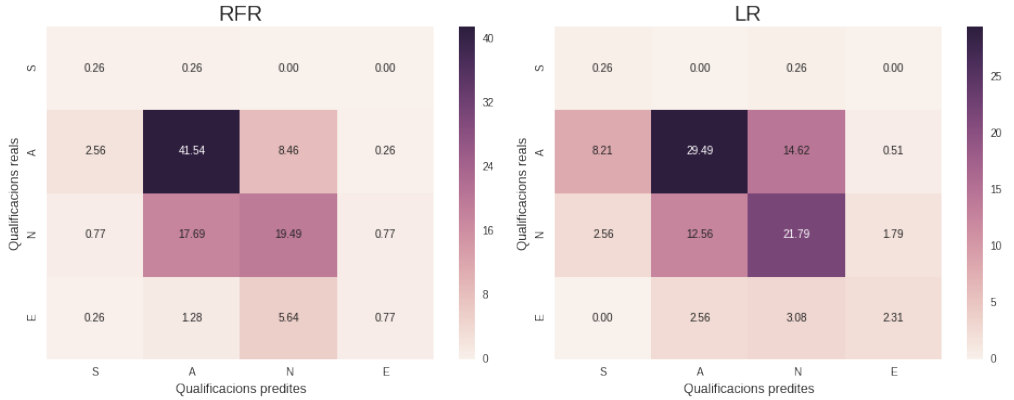
En aquestes proves podem seguir veient com els millors predictors són el RCxE i el RFR, tot i que ara ambdós presenten més mostres atípiques. Les distribucions i els resultats són semblants als resultats provats amb les dades anteriors. L'únic predictor que marca més diferència és el Random Forest Regressor, ja que disminueix força la mediana de la distribució, i l'error promig absolut.

Ara passem a veure els resultats que s'han obtingut amb les notes discretes (suspés, aprovat, notable i excel·lent). Com ja s'ha explicat, la representació d'aquestes proves es fa amb una matriu de convolució, seguidament paso a descriure que representa.



(a) RCxE

(b) RCxA



(a) RFR

(b) LR

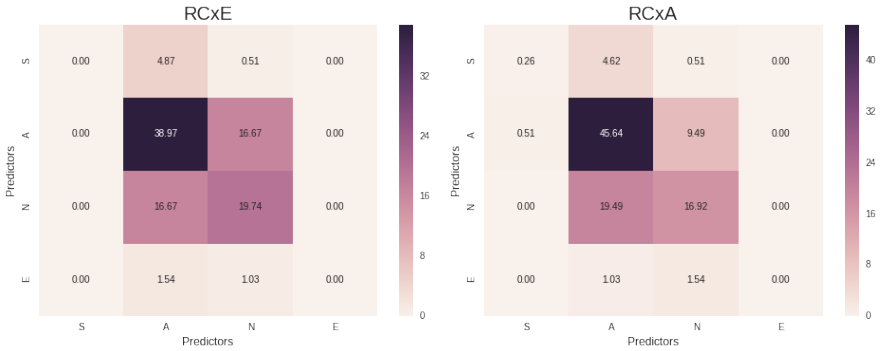
Figure 31: Matriu de convulsió. Training: primer

En cada matriu de colors podem veure 16 caselles, cada fila i columna posa: S, A, N i E, que significa: suspés, aprovat, notable i excel·lent. Cada casella significa el percentatge de vegades que la qualificació predita coincideix amb la qualificació real corresponent. Possem un exemple per acabar d'entendre, en la primera figura apareix en la correspondència de aprovats amb aprovats un 40,26%, això significa que el 40,26% de les proves diu el predictor que un alumne treurà una nota d'aprovat i la nota real era aprovat. Si ens fixem com més alt sigui el percentatge, més fosca és la casella. Per tant el que

busquem és tenir fosca la diagonal, ja que vol dir que ha encertat tot. Ho representem amb aquest tipus de representació pk equivocar-se de suspés a aprovat és acceptable, però no de suspés a excel·lent, és per això que s'ha de veure una tendencia de color a les rodalies de la diagonal.

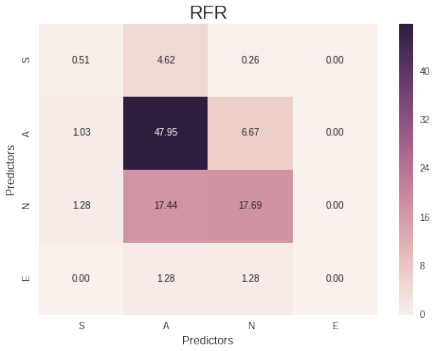
Podem veure que els millors predictors són els recomanadors, ja que mantenen força alta la diagonal i el color fosc es manté força concentrat a la diagonal. Si ens fixem en el *Random Forest Regressor* i en el *Regressor lineal* confon més per les seves rodalies. En canvi trobem que tots els predictos confonen aprovats per excel·lents.

Per un altre banda també s'ha provat amb *training* les notes de primer juntament amb les de segon i com a *test* les notes de tercer.

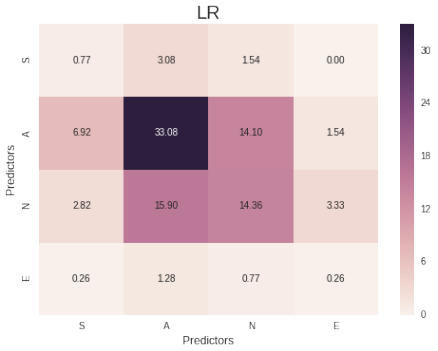


(a) RCxE

(b) RCxA



(a) RFR



(b) LR

Figure 33: Matriu de convulsió. Training: primer i segon

En aquesta figura es pot veure com RCxE, RCxA i RFR tenen resultats semblants, però en canvi veiem que el regressor lineal es dispersa molt i sol

fer fallos amb un salt de dos categories, com per exemple confondre notables amb suspesos. Tot i així, tots ells concentren força bé els resultats a la diagonal.

Per finalitzar la part de proves quantitatives, ens queda fer les proves per al ranking d'assignatures. Per això s'ha utilitzat, com ja s'ha explicat, la mesura de *Mean Ranking Score* per avaluar la qualitat de cada predictor i veure quin predictor és més bo per fer un ranking. Els resultats obtinguts són purament numèrics i venen representats en la següent taula:

	RCxE	RCxA	RFR	LR
MRS	2.05	2.975	2.2	1.975

Table 6: Mean Ranking Score. Training: primer

	RCxE	RCxA	RFR	LR
MRS	2.375	3.4	2.375	2.525

Table 7: Mean Ranking Score. Training: primer i segon

Hem mostrat les dues proves, amb diferents *training*, juntes per poder-les contrastar. Recordem que la mesura MRS és una mesura que com més propera a 0, millor. Si ens fixem els valors són més baixos amb la primera taula, i el millor predictor per aquesta és el recomanador col·laboratiu basat en l'estudiant. On el *training* es fa amb les notes de primer i segon podem veure que els millors predictors són RCxE i RFR.

Proves qualitatives

Un cop hem fet les proves quntitatives i s'ha obervat quin predictor és millor per cada cas, passem a utilitzar el millor predictor per veure un cas d'èxit i un de fracàs. En aquest apartat mostraré un cas d'èxit i de fracàs per fer una recomanació de notes, i seguidament un cas d'èxit i fracàs del ranking d'assignatures.

En les proves quantitatives, hem pogut veure que el millor predictor per predir notes quantitatives (de 0 a 10) és el recomanador col·laboratiu basat en l'estudiant. És per això que utilitzarem aquest predictor per fer les proves qualitatives.

6 Conclusions i treball futur

7 Bibliografia