



Treball final de grau

GRAU D'ENGINYERIA
INFORMÀTICA

Facultat de Matemàtiques
Universitat de Barcelona

Sistema intel·ligent de suport al Tutor d'estudis

Autor: Xavier Moreno Liceras

Directora: Laura Igual

Realitzat a: Departament

Matemàtica Aplicada y Anàlisis

Barcelona, June 8, 2015

Abstract

Goldbach's weak conjecture asserts that every odd integer greater than 5 is the sum of three primes. We study that problem and the proof of it presented by H. A. Helfgott and D. Platt. We focus on the circle method. Finally, we describe a computation that confirms Goldbach's weak conjecture up to 10^{28} .

Resum

Normalment un tutor tutoritza a un conjunt d'alumnes i no dóna temps de mirar detingudament alumne per alumne, això fa que possiblement no es realitzin les accions corresponents per a un alumne.

Ha de
ser
redac-
tat en
primer
per-
sona del
present

Revisió
de les
faltes
d'ortografia

Agraïments

Vull agrair a ...

Contents

1	Introducció	1
2	Descripció del problema	2
2.1	Explicar dades	2
2.2	Ciència de les dades	2
2.3	Etapas del projecte	3
2.3.1	Preguntes	3
2.3.2	Adquisició	3
2.3.3	Neteja de dades	3
2.3.4	Clusterització	4
2.3.5	Predicció	4
2.3.6	Evaluació	4
2.4	Preguntes plantejades	5
3	Planificació	7
3.1	Tasques	7
3.2	Diagrama de Gantt	7
3.3	Evaluació econòmica	7
4	Desenvolupament del projecte	8
4.1	Eines	8
4.1.1	Eines de suport	8
4.1.2	Eines de programació	9
4.1.3	Eines d'edició	10
4.2	Tècniques utilitzades	11
4.2.1	Clusterització (Agrupacions)	11
4.2.2	Predicció	14
4.2.3	Reducció de dimensions	18
5	Experiments i resultats	19
6	Conclusions i treball futur	20
7	Bibliografia	21

1 Introducció

2 Descripció del problema

2.1 Explicar dades

change
title

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 àrees, 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. **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*)
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 addicionals)
4. **Descobrimet** 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, classificadors, recomanadors)
7. **Evaluació** Com de segur estem dels nostres resultats? (Proves estadístiques, rendiment del model)

2.3 Etapes del projecte

2.3.1 Preguntes

La primera etapa va ser el plantejament de les preguntes que volíem resoldre. A partir de la plataforma trello, entre els participants del projecte vam plantejar preguntes, les quals entre tots decidíem amb quines preguntes ens quedariem i respondríem. Moltes de les preguntes no podíem saber si les podíem respondre fins que ens arribessin les dades, ja que depeníem totalment de la informació que contenien les dades.

2.3.2 Adquisició

L'adquisició de les dades va ser a partir del Vicerectorat de Política Docent. Les dades ens van arribar 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 he hagut de netejar les dades per tal de poder treballar amb elles. Aquestes van ser les netejes que vaig fer:

Canvi de format Com ja he explicat abans les dades ens van arribar en forma de fulla de càlcul, on en cada fulla havia una taula amb diferent informació. Per poder manipular-les millor des de Python, vaig haver de separar cada fulla en un fitxer amb format *csv*, de tal manera que va quedar un fitxer *csv* per taula.

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 hem pogut adquirir el curs i semestre que es cursa cada assignatura, per tant el que faig és creuar aquestes dades amb les dades que tinc 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 gent que s'havia matriculat aquest any 2014, però encara no havien acabat de cursar l'assignatura, en aquesta els hi apareixia un 0. Això feia que dintre de les notes dels alumnes haguéssin dades incoherents, per aquesta raó vam decidir eliminar totes les notes del segon semestre i de l'any 2014 que són un 10.91% de les notes.

Elimino
o no al
final?

Normalització 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, ... vam decidir normalitzar les notes per any i per assignatura aplicat 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 he 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 construïda la predicció, hem d'avaluar quant de bona és. Per això vaig agafar un 10% de les meves dades per poder testejar i comprovar la taxa d'encerts de la predicció.

2.4 Preguntes plantejades

change
title

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. He agafat tots els alumnes que hagin cursat totes les assignatures de primer i després amb 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 taxa 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 cauen al perfil d'alumnes que ho suspenen tot són els que solen abandonar la carrera. Fins ara això sembla força obvi, però ho podem 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 venen d'un Cicle Formatiu de Grau Superior (CFGs) solen ser alumnes que els hi dona malament les assignatures relacionades amb les matemàtiques i són força bons en programació. Ara bé, això és cert? Per això ens plantegem aquesta pregunta, a partir de perfils d'origen, volem saber amb quin cluster destí van a parar. 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
2. Cicle Formatiu de Grau Superior
3. Diplomats, Llicenciats

4. Salt d'Universitat

Predicció de notes per fer un ranking de dificultats

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 anirà fluix per així pogui reforçar més el temari que es donarà en aquella assignatura. Recordem que la finalitat d'aquest projecte es 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 d'aquesta eina en decidirà que fer.

3 Planificació

3.1 Tasques

3.2 Diagrama de Gantt

3.3 Evaluació econòmica

4 Desenvolupament del projecte

4.1 Eines

4.1.1 Eines de suport

Aquestes són les eines de suport que m'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'emmagatzement 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. Nosaltres 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 en el seu moment per dues simples 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 agregació 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 és una biblioteca informàtica orientada a l'aprenentatge automàtic per a Python. Té suport per classificadors, regressors i clustering. 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 visualització 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 sns
palette = sns.color_palette("hls", 5)
sns.palplot(palette)
```



Figure 1: 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'empren 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 escollir aquest entorn d'edició ja que en un projecte de ciència de les dades s'han de veure resultats 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 podem 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 si el que volem es 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 tècniques 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, segmenta aquests n elements en k grups (entrada obligatòria 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, \dots, 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 $S = \{S_1, S_2, \dots, S_k\}$. Amb la finalitat de minimitzar la suma dels quadrats dintre de cada grup:

$$\arg \min_S \sum_{i=1}^k \sum_{x_j \in S_i} \|x_j - \mu_i\|^2$$

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

Com es veu en la fórmula, 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 apliquem iteracions sobre aquesta fórmula.



Figure 2: Some caption

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 3: Some caption

Mètriques utilitzades

Existeixen dos indicadors d'avaluació dels resultats de l'anàlisi de les agrupacions:

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*.

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.

Referenciar
a Sil-
houette
de
sklearn.

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 he utilitzat per aquest projecte.

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.

linar
la teo-
ria de
TNUI -
recom-
menders

Imaginem que tenim una matriu tal que:

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

on:

e_i és un estudiant.

a_i és una assignatura.

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

? 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 posar la nota més adient. Abans d'explicar com funciona, introduiré els diferents tipus de recomanadors que podem tenir:

Recomanador col·laboratiu basat en estudiant 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 semblant a un alumne e_j , les assignatures que no ha cursat e_i podem dir que seran semblants a les notes que ha tret e_j en aquelles assignatures.

Recomanador col·laboratiu bassat en assignatures 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. Aquest recomanador no s'ha fet servir en aquest projecte, però es podria fer servir si es pogués aprendre 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:

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

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. Realment aquesta fórmula és la fórmula de 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

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 conjuncions de característiques que han portat fins a la fulla.



Figure 4: Some caption

Bé doncs, *Random Forest Regressor*, no és més que 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

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 dependent a partir d'un conjunt de variables, és a dir:

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

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



Figure 5: Some caption

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. En aquest hem utilitzat tant mètriques supervisades com no supervisades.

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:

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

Coefficient de pearson (PCC) El coeficient de pearson, la utilitzem com una mètrica supervisada i la fem servir per diferencia 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) Per últim també calculem la desviació estàndard per veure si els errors són més o menys dispersos. La fórmula utilitzada és la següent:

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

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 reduir el cost computacional sense variar en el seu resultat. Una de les tècniques utilitzades en aquest projecte és l'anàlisi de components principals (PCA).

[link de
sklearn](#)

PCA

L'anàlisi 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 i 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

<i>Algoritme\Mètriques</i>	MAE	MSE	PCC
Recomanador col·laboratiu	1.231	2.997	0.335
Recomanador basat en contingut	1.197	2.905	0.403
Random Forest Regressor	1.134	2.584	0.490
Linear Regressor	1.175	2.720	0.462

Table 1: Dades no normalitzades

<i>Mètriques/Algoritme</i>	MAE	MSE	PCC
Recomanador col·laboratiu	0.558	0.669	0.069
Recomanador basat en contingut	0.531	0.660	0.358
Random Forest Regressor	0.509	0.565	0.393
Linear Regressor	0.538	0.648	0.462

Table 2: Dades normalitzades

6 Conclusions i treball futur

7 Bibliografia