Universitat Politècnica de Catalunya

Facultat d'Informàtica de Barcelona

Speed Dating

Aprenentatge Automàtic

Narcís Terrado i Yaiza Cano

Gener 2021

Contents

1 Introduction

L'objectiu principal d'aquest treball és desenvolupar un model de classificació sobre un problema a escollir utilitzant diferents mètodes linears i no linears i concloure quin quines són les característiques del millor model pel tipus de dades proporcionat.
Les dades [1] que hem triat i que utilitzarem per entrenar el nostre model han estat recaptades per participants d'esdeveniments de cites ràpides entre els anys 2002 i 2004 a Estats Units. Durant aquests esdeveniments, cada participant té una cita de només 4 minuts amb cada participant del sexe oposat. A mesura que avancen els minuts, els participants van emplenant un qüestionari amb informació relacionada amb els seus gustos personals i la valoració dels diferents atributs de l'altra persona per, finalment, valorar si la volen tornar a veure en una segona cita.

Aquest *dataset* està format inicialment per:

- 123 atributs on hi ha tant numèrics com categòrics.
- 18372 *missing values*.
- 8378 instàncies/mostres del problema.

Així doncs, el nostre objectiu és entrenar un model fiable que sigui capaç de, donat un nou qüestionari de cita, concloure si hi haurà segona cita o si el/la participant dirà *next*.

2 Related Work

La nostra inspiració a l'hora d'escollir aquest *dataset* per dur a terme la pràctica ve donat, apart pel fet que són unes dades divertides amb les que treballar i amb les que, donat els anys entre els quals estan recollides i el lloc, vam coincidir en que podríem obtenir resultats controversials per avui dia, també vam trobar que hi havia treballs molt bàsics fets amb aquestes dades i vam decidir aprofitar l'oportunitat de fer-ne un de ben fet.

El primer treball que volem mencionar l'ha realitzat *Keith McNulty* [2] i es titula *What Matters in Speed Dating?* [3], títol particularment graciós donat que, de les 123 variables que conté el *dataset* inicialment, decideix *dropejar* la gran majoria i quedar-se només amb 15. Una mica esbiaixada la seva opinió sobre què influeix a les cites ràpides, no?
Això és tot el que sabem del *pre-processing*. Intens, oi?
En quant als mètodes que aplica, només n'aplica un de lineal, el *logistic regression*. Si més no, l'elecció d'aquest mètode té sentit per nosaltres posat el tipus de dades amb les que s'ha quedat (principalment característiques d'una persona, com ara la sinceritat, si és graciosa, intel·ligent, etc.) i que el seu objectiu és

saber quin pes té cadascuna a l'hora de decidir si tenir una segona cita o no.
De l'*output* que proporciona el seu model, es fan les següents observacions:

- L'opinió del participant sobre les característiques d'una persona és el principal indicador sobre si hi haurà segona cita. (Sorpresa per ningú).
- D'entre les diverses característiques, l'atractiu sembla ser el més substancial.
- Ser sincer i ambiciós són qualitats que fan decréixer la probabilitat de fer *match*. L'autor apunta que semblen ser *turn-offs* per les cites potencials. Aquest fet sí que ens va sorprendre i tenim moltes ganes de contrastar-lo amb els nostres models.
- La resta de factors prenen un rol menor positiu.

Per últim fa una comparativa entre les persones que s'han identificat amb gènere femení i les que s'han identificat amb gènere masculí. Nosaltres també trobem que és interessant donat que en aquest estudi només s'ha considerat la preferència sexual heterosexual.
El seu estudi destaca dues diferències principals:

- Seguint els estereotips, els homes semblen donar molta més importància a l'atractiu físic mentre que les dones l'hi donen a la intel·ligència.
- El fet que es coneguessin d'abans, tenen un pes important per ambdós grups. Els homes prefereixen una cara nova mentre que les dones prefereixen una cara coneguda.

Per últim menciona que l'estudi està notablement acotat per les variables, que el *dataset* és molt més gran i que hi ha molt a explorar. Aquestes paraules les hem pres com una invitació. Ens acompanyes en aquesta, la nostra *estimada dataventura*?

3 Data Exploration

En aquest apartat comentarem les transformacions que s'han fet a les dades des de que es van recollir de la base de dades *OpenML* [1] fins que vam decidir que estava correcta per entrenar els models.

3.1 Basic Inspection

Fent una primera inspecció molt bàsica de les dades ens vam adonar de dos inconvenients.

Els atributs que els participants no van contestar estaven marcats amb un '?'. Aquest caràcter no és interpretat correctament per *numpy* ja que aquesta llibreria ja té la seva pròpia manera de marcar caselles que no han estat contestades,

així doncs, vam substituir tots els '?' per *NaNs*.
Numpy interpretava el tipus de la majoria dels atributs com a *object*. Això trobàvem que era un inconvenient ja que a l'hora d'inspeccionar les mostres, no obteníem informació numèrica dels atributs corresponents. Així doncs, vam procedir a assignar un tipus (categòric o numèric) a cada atribut de les dades.

3.2 Data Cleaning

Les dades escollides contenen moltes files i columnes que considerem innecessàries per diverses raons i que hem decidit eliminar.

Els primers dos atributs: *'has_null'* i *'wave'* ens aporten informació sobre les dades que hi trobarem en aquella instància però no sobre com ha anat la cita directament ni influeixen en el resultat de si hi ha hagut *match*.

Seguidament, vam fer una passada manual sobre els atributs i vam debatre quines eren les columnes que aportaven informació rellevant i quines no. En general vam concloure que hi havia molta informació representada varis cops de diferents maneres; per exemple, es demana tant a la parella com al participant que valorin, del 0 al 100, quan d'important és per cadascun d'ells que la persona amb la que tenen una cita sigui atractiva (*pref_o_attractive* i *attractive_important* respectivament) i, posteriorment es representen aquests mateixos valors en grups d'intèrvals (*d_pref_o_attractive i d_attractive_important respectivament)*. Creiem que la informació que pot influir d'una manera més clara i directa és la numèrica però, per motius que expliquem a la secció de *report* 8, hem decidit quedar-nos amb els intervals.

A continuació mirem la quantitat de *NaNs* que conté cada columna i fila. Per cadascuna d'elles computarem el percentatge de *NaNs* que hi trobem i eliminarem aquelles que considerem que tenen un percentatge massa elevat per imputar i que les dades siguin fiables i tornarem a revisar i qüestionar la utilitat de les variables restants.
Així doncs, donarem un exemple del procediment que hem aplicat: donats els percentatges de les columnes, el valor més elevat que trobem és de 78.515159% per la variable *expected_num_interested_in_me*, el següent valor és de 14.000955% i correspon a la variable *expected_num_matches*. Tot i que inicialment puguem pensar que el segon valor no és gaire elevat i que encara podríem treballar amb aquestes dades, és més que evident que el primer és inviable. Tot i així vam decidir eliminar també la segona variable degut a que consideràvem que no aportava informació rellevant.

3.3 Data imputation

En aquest moment del *pre-processing* tenim un *dataset* format per 68 variables i 8378 instàncies. De totes aquestes dades, encara tenim valors amb els que no podem estimar un model, aquestes dades són els *NaNs* que han restat del

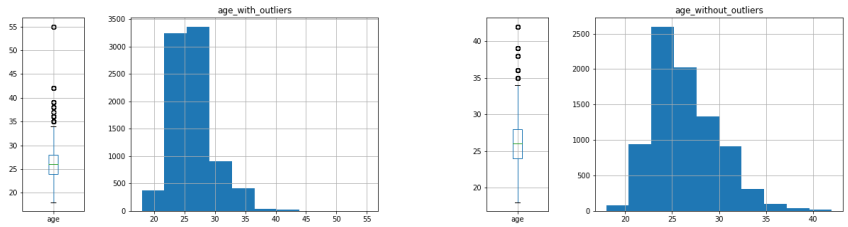
procediment anterior. Aquests valors s'han de tractar i substituir per uns altres que siguin vàlids i que no alterin notablement les mostres de manera que el resultat que s'obtingui del model entrenat amb aquestes dades sigui el més fiable possible al que s'obtendria amb unes dades ideals.

En quant als valors numèrics, hem intentat imputar-los utilitzant el classificador *K nearest neighbors*. Aquest classificador però, no ens ha donat bons resultats degut a que només teníem un únic atribut numèric que no tingués *NaNs* amb el que entrenar-lo. Així que hem decidit substituir per la mediana ja que com que encara no hem fet el tractament d'*outliers*, usar la mitjana hagués pogut influenciar negativament als valors imputats.

Per les variables categòriques hem usat la moda per substituir els *NaNs*. Hem escollit la moda ja que al no tenir un percentatge molt elevat de *NaNs* a cap columna, la moda es un valor suficientment bo que no tindrà impacte negatiu més endavant.

3.4 Finding Outliers

Les variables numèriques d'aquest *dataset* estan acotades així que, provant de trobar *outliers* visualitzant *boxplots* i aplicant *local outlier factor* sabíem que no trobaríem res. Tot i així, vam decidir aplicar-ho a les variables que ens havien donat problemes al imputar *NaNs* amb *Knn*, és a dir, les variables *age* i *age_o* i hem pogut descartar les cites que ha tingut la persona de 55 anys.
Adjuntem un gràfic 1 on es pot apreciar el que acabem de comentar, les gràfiques per la variable *age_o* és exactament igual.



(a) amb *outliers* (b) sense *outliers*

Figure 1: Valors de la variable *age* amb i sense *outliers*

Després de l'extracció de mostres que contenien l'edat esmentada, ens roman un *dataset* format per 69 variables i 8366 instàncies.

3.5 Encoding Categorical Features

Per tal de transformar les dades en un rang de valors que pugui acceptar un estimador de *machine learning*, hem decidit aplicar tractaments diferents segons

les característiques de les nostres dades.

Per les variables categòriques que defineixen intervals, hem decidit aplicar *ordinal encoder*. El motiu d'això és degut a que la representació numèrica que aquest codificador ofereix té en compte una entrada continua i interpreta les dades com ordenades.
Tot i així sabem que les dades obtingudes després d'aplicar aquest codificador poden no ser aptes per utilitzar-les directament a alguns estimadors. Per tant, caldrà normalitzar-les o estandarditzar-les.

En quant a la resta de variables categòriques, hem decidit aplicar *one hot encoding*. Després d'aplicar aquest procediment se'ns han afegit noves columnes al *dataset* i n'hem esborrat d'altres. A partir d'ara, i aquest serà el tamany final, treballarem amb 76 variables i 8366 mostres.

3.6 Normalization

Per últim, cal normalitzar totes les variables, hem escollit utilitzar *minmaxscaling* degut a que les variables categòriques que no representaven intervals ja es trobaven acotades entre els rangs [0,1]. Hem optat per usar aquest *scaler* ja que tenim poques variables numèriques, la majoria són variables categòriques que ja hem codificat.

4 Resampling Protocol

Arribat a aquest moment del projecte on, per fi, començaríem a entrenar models i a veure si no ens havíem carregat gaire el nostre *dataset*, volem comentar una experiència que, en el seu moment vam patir molt però que ara ens sembla fins i tot divertida.

La nostra intenció sempre ha estat provar varis protocols de mostreig amb els diversos models que es demanen. Així doncs, vam provar *k-fold cross-validation* amb varies k, vam provar $LOOCV$ i un munt de tamanys de *train_test_split* que, per nosaltres, tenien sentit. Provéssim la combinació que provéssim, els models sempre assolien una *accuracy* del 100%. No es que tinguem falta de confiança en nosaltres mateixos, però no ens quadrava que haguéssim fet un *pre-processing* ideal i que les nostres dades fossin increïblement explicatives i imprescindibles. Inicialment pensàvem que no estàvem dividint bé les dades o que no estàvem aplicant bé els mètodes. El problema era que havíem mantingut les variables de *decision* i *decision_o* amb les quals es computa la variable a predir *match*. Un cop més, es demostra que la màquina guanya a l'humà, sobretot si aquesta és una gandula que no vol computar més de lo necessari.

Finalment, ens hem decidit usar *k-fold cross-validation* a través del model de sel·lecció d'hiperparàmetres *GridSearchCV*. El rang de k és [2, 14] i els hiper-

paràmetres que es proven depenen del model i els comentarem a l'apartat corresponent de resultats. El nostre objectiu utilitzant aquesta búsqueda exhaustiva és trobar la combinació de paràmetres i de tamanys de *train* i *test* que millor funcioni per a cada classificador per després analitzar els coeficients que proposen.

5 Results using Linear/Quadratic Methods

Hem entrenat 3 models linears. Per a tots ells, tant els valors dels hiperparàmetres com els tamanys dels k *subsets* són els que millors resultats han proporcionat després d'aplicar el mètode de sel·lecció explicat a l'apartat anterior.

A continuació, per cada model entrenat mostrarem els paràmetres que hem utilitzat, l'*accuracy* i l'error obtinguts i, en quant als coeficients estimats, mostrarem els top 5 millors i pitjors i veurem, en mitjana, què aporten la resta de paràmetres.

5.1 Linear Discriminant Analysis

La combinació de paràmetres que maximitzen l'*accuracy* corresponen a $k = 6$ i són els següents:

shrinkage	solver	accuracy	mean squared error
None	lsqr	84.72%	0.1528

Table 1: LDA Performance

A partir del *GridSearchCV* i provant un petit set de paràmetres hem escollit els que el GridSearchCV ens ha donat com a millors paràmetres. En un principi havíem pensat que al tenir tantes columnes, hauríem d'usar el *shrinkage* que ens proporciona el mètode donat que el dataset conté moltes *features*, però el resultat del *GridSearchCV* ens va indicar el contrari, així doncs, no usem *shrinkage*. Com a solver, usem *least squared*, tot i que de nou, pensavem que hauríem d'usar *svd* pel nombre de *features*. Potser tenim la sensació de que tenim moltes *features* per la poca experiència que tenim treballant amb *datasets*.

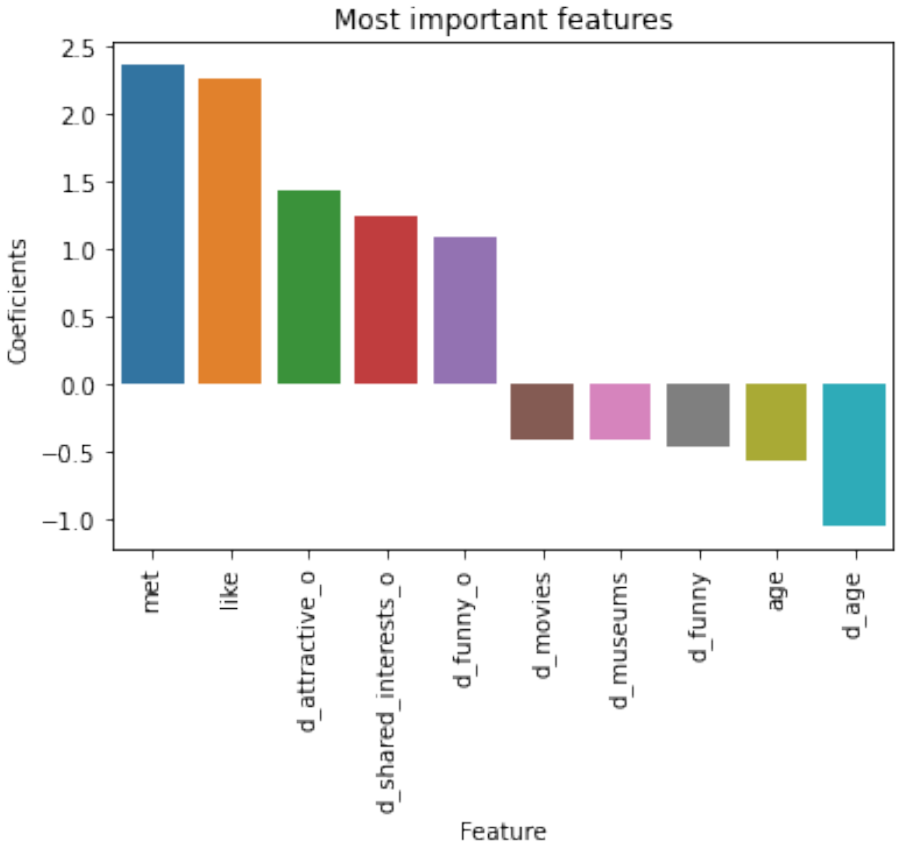


Figure 2: 10 *features* més rellevants de LDA

A la figura 2 podem veure-hi els pesos de les 10 *features* més rellevants. Per una banda tenim les variables que influeixen positivament en la decisió de fer *match*: la variable *met*, una variable binaria que indica si les dues persones de la cita es coneixien prèviament. En segon lloc hi ha la variable *like* que en el *dataset* és un natural del 0 al 10 que indica quant li ha agradat l'altre persona al participant. Teníem la hipòtesi que aquesta anava a ser una *feature* important, ja que per sentit comú, com més t'agradi una persona, més probabilitat hi ha de fer *match*. Les altres 3 *features* importants són *d_attractive_o*, *d_shared_interests_o* i *d_funny_o* que indiquen com d'atractiva considera a l'altra persona al participant.

De les *features* que influeixen negativament, veiem que la diferència d'edat *d_age* té el pes més important, que indica que a més gran la diferència d'edat, menys probabilitat hi ha de fer *match*. Aquesta representació la veiem acompanyada amb la variable edat, *age*, com més anys tingui una persona, més li costarà fer

match. Aquesta afirmació però, hem d'esclarir, que es dona en aquest *dataset* degut a que la mitja d'edats és de 27, si la mitja d'edat fos superior, probablement aquesta *feature* no seria tan perjudicial. Les últimes tres variables tenen a veure amb la personalitat del participant i ens han sorprès notablement però, que no s'estengui el pànic, tenim teories per a totes com a bon amants del drama que som. El fet de que el participant sigui una persona amb sentit de l'humor, *d_funny*, pot fer que la parella associi aquesta característica amb la de ser infantil i/o poc madur o que el mateix participant busqui algú que es rigui de la vida i que no visqui amargat. En quant a que el participant sigui un amant dels museus, *d_museums* pot fer que la parella pensi que és avorrida o massa seriós. Sent sincers, en quant a que el participant sigui un cinèfil, *d_movies*, no li trobem explicació, potser la majoria va debatre quin gènere de pel·licules era millor i no van acabar coincidint.

Com a conclusió podem extreure que: les persones que estan entrant en una edat ja més avançada, haurien de buscar l'amor en un altre tipus d'esdeveniments i programes com ara *first dates*; que tot i que a la parella li importi molt que es comparteixin interessos, si no li agraden les mateixes pel·lícules que al participant, ho té bastant cru; que si ja es coneixien d'abans, ja tenien molt recorregut fet i és més fàcil conèixer coses no tan típiques i comunes que es poden trobar interessants i atractives.

5.2 Logistic Regression

La combinació de paràmetres que maximitzen l'*accuracy* corresponen a $k = 13$ i són els següents:

C	penalty	solver	accuracy	mean squared error
1	l1	liblinear	86.0%	0.1399

Table 2: Logistic Regression Performance

De nou, hem provat un set de paràmetres amb *GridSearchCV* per a saber quins podien ser els millors. El paràmetre de regularització *C*, com més proper a 0 és, més fortament regularitzat és, i més *underfitting* es crea. Per a valors més grans de *C*, reduïm la regularització i per tant, creem un model més complexe amb risc de *overfitting*. S'ha escollit una *C* que es troba a un punt mig, ni molt gran ni molt petita, el que ens porta a pensar que no hi haurà *overfitting* ni *underfitting*. Per el paràmetre de *penalty*, tenim que *GridSearchCV* ha trobat que *l1* és el més optim. La diferència entre els dos és que amb *l1*, es fa un *shrinking* de les *features* menys importants. Això provoca que es faci una mena de selecció de les *features* en cas de que en tinguem un nombre elevat, com és en el nostre cas, per això s'ha escollit aquest paràmetre. Donat que hem escollit com a *penalty l1*, usem el *solver* que maneja millor aquest tipus de regressió, que quan usem *l1* se'n diu *Lasso Regression*.

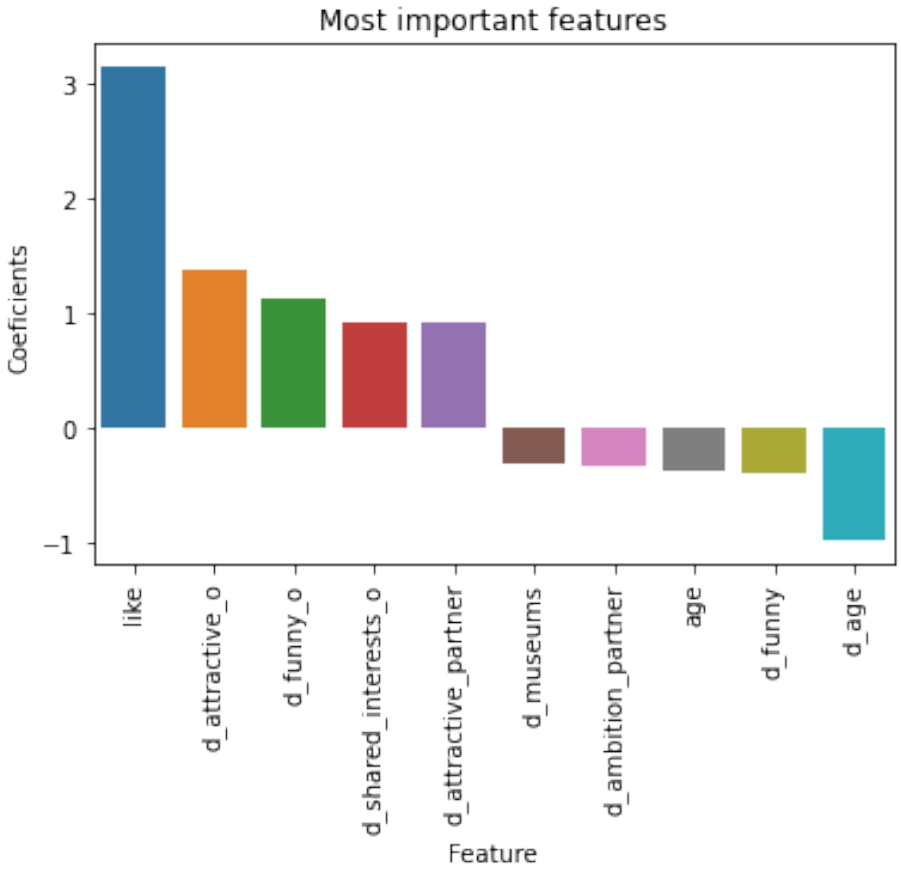


Figure 3: Les 10 *features* més rellevants de Logistic Regression

Veiem que amb la *Logistic Regression* les *features* que influeixen positivament en la decisió final són les mateixes que prediu *LDA* amb l'intercanvi de la variable *met* amb la *d_attractive_partner*. Aquesta última té sentit per nosaltres donat que és important que trobis atractiva a una possible parella en una cita de només 4 minuts on, aspectes com ara la personalitat i els objectius a la vida poden no donar temps a lluir-se.

En quant a les variables que influexien negativament, també veiem les mateixes que al classificador *LDA* amb l'intercanvi de *d_movies* per *d_ambition_partner*. Això ja ens sona millor a nosaltres ja que com hem explicat abans no trobàvem sentit a que hi hagués un problema en que agradi el cine. En quant a l'ambició, una mica està bé però tots sabem que els hi passa a les persones massa ambicioses.

5.3 Linear SVM

La combinació de paràmetres que maximitzen l'*accuracy* corresponen a $k = 9$ i són els següents:

C	loss	penalty	accuracy	mean squared error
2	squared_hinge	l2	84.9%	0.1507

Table 3: Linear SVM Performance

De la mateixa manera que a la *Logistic Regression*, el paràmetre C és el paràmetre de regularització, com més elevat, més llibertat té el model, menys regularitzat està, i com més propera a 0, més restringit i més regularitzat. El *GridSearchCV* ens ha donat coma millor valor $C = 2$, per tant, aquest model té més llibertat que el que hem usat per a la *Logistic Regression*. Per a la *Loss function* s'ha escollit la *squared_hinge* ja que és la única compatible amb el paràmetre de *penalty*.

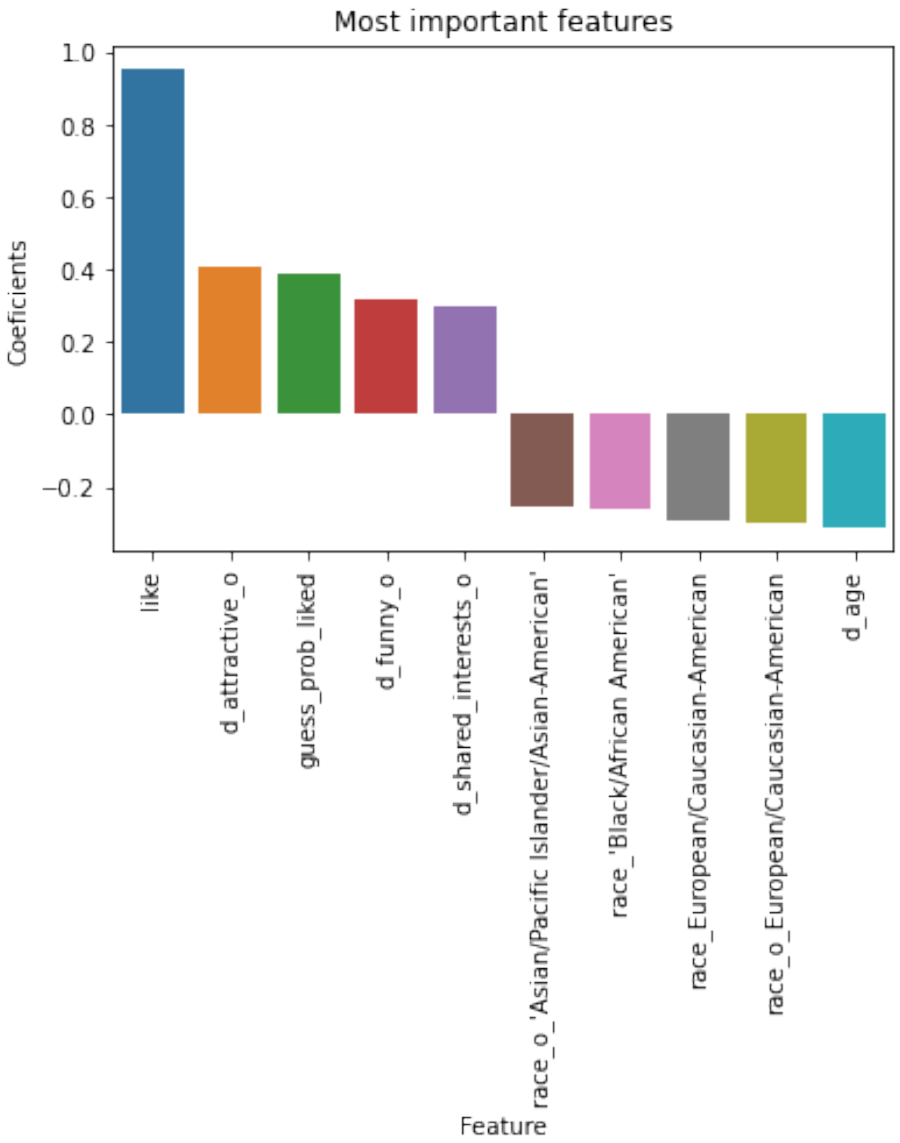


Figure 4: Les 10 *features* més rellevants de Linear SVM

El model *Linear SVM* és, sense dubte, el més original de tots. Si ens fixem en les variables que influeixen positivament, veiem que són les mateixes que estima el primer classificador amb l'intercanvi de la *feature met* per *guess_prob_liked*. Aquesta variable té sentit per nosaltres posat que si sents que la teva parella es troba interessada en tu i que la conversa flueix, tot i que sigui durant 4 minuts, la sensació de que hi ha *feeling* deixa al cos en un estat de felicitat i benestar

molt particular.

En quant a les variables que influeixen més negativament, aquest model llança la casa per la finestra, renega del que diuen els seus companys i aposta per un problema racial. Serem sincers i admetrem que quan vam veure el *dataset* per primer cop i tenint en compte els anys durant els quals es va recaptar la informació i el lloc, també vam apostar per això. El que no ens esperàvem era que les pitjors races per participar en un esdeveniment de cites ràpides entre 2002 i 2004 a EEUU eren totes menys ser *latino/hispanic amèrican* o d'una altra raça no contemplada en el qüestionari. Si mirem la quantitat de gent que hi havia de cada raça, podem veure que la gran majoria de gent era european/caucasian-america, així que podem suposar que realment no és cert el que acabem de dir de les pitjors races (ja que aquestes no tenen suficient representació) però volíem comentar-ho per veure si creava controvèrsia en una primera lectura. La veritat és que no entenem d'on surten aquests coeficients, d'entre nosaltres 3, està clar que aquest classificador és la *drama queen* per excel·lència.

5.4 Performance Comparative

Visualitzem totes les *accuracy* juntes.

Model	Accuracy	mean squared error
LDA	84.5%	0.1549
Log Reg	86.0%	0.1399
Linear SVM	84.9%	0.1507

Table 4: Linear/Quadratic Performance Comparative

Podem observar que el model amb més precisió ha estat ella *Logistic Regression*, tot i que tots els models han estat molt a la par, i la diferència entre la precisió pot haver estat donada per factors aleatoris amb la partició del dataset en *train* i *test*. Tots els models han tingut unes *features* rellevants molt semblants. El model més diferent en aquest aspecte ha estat el *Linear SVM*, en el que hi hem trobat les races com a coeficient negatiu, cosa que no hem trobat en cap altra. El que ens pot indicar el fet de que tots els models hagin tret unes *features* rellevants semblants es que aquelles *features* son realment importants a l'hora d'escollir parella.

6 Results using General Non-Linear Methods

Hem entrenat 2 models no linears. Per a ambdós d'ells, els valors dels hiperparàmetres són els que millors resultats han proporcionat després d'aplicar el mètode de sel·lecció explicat a l'apartat anterior. Com a valor de k s'ha utilitzat 5 degut a que provar-ne varis com als mètodes linears costava molt de temps i a Internet deia que era el més utilitzat.

A continuació, per cada model entrenat mostrarem els paràmetres que hem utilitzat, l'*accuracy* i l'error obtinguts.

6.1 SVM with RFB kernel

C	gamma	accuracy	mean squared error
100	0.01	85.7%	0.1428

Table 5: SVM with RFB kernel performance

C és de nou el paràmetre de regularització del model. A partir del *GridSearchCV* s'ha trobat que per aquest *dataset* el millor valor de *C* provat era de 100, un valor molt més elevat que en els models lineals. Això ens indica que el model té més llibertat i és mes complexe, així que patim risc de *overfitting*. El paràmetre *gamma* indica la influència de cada *sample*. En el nostre cas, *GridSearchCV* ha trobat un valor de *gamma* bastant baix, de manera que pot ser que el model sigui massa restringit i no pugui capturar la complexitat de les dades. Però al tenir un valor de *C* elevat, pot ser que els dos valors es compensin per aconseguir la precisió obtinguda.

En el cas de SVM amb kernel RBF, no tenim l'extracció de coeficients com en els altres models donat que al no ser un mètode lineal, els pesos de les *features* són més complicats d'obtenir.

6.2 Random Forest

La combinació de paràmetres que maximitzen l'*accuracy* són els següents:

max depth	max features	nºestimators	accuracy	mean squared error
25	sqrt	200	85.4%	0.1458

Table 6: Random Forest Performance

El paràmetre més important és *n_estimators*, que indica el nombre d'arbres del *Random Forest*. Segons tenim entés, un major nombre d'arbres implica una major rendiment a canvi d'un cost computacional més elevat. En el nostre cas, hem tornat a fer ús de *GridSearchCV* per a determinar els millors paràmetres i hem provat el paràmetre de *n_estimators* amb un rang de 10 a 2000 i el millor resultat ha estat el de 200, creiem que és degut a que per el nostre *dataset* ja n'hi ha prou.

En el cas de *max_features*, el *GridSearchCV* ens ha escollit *sqrt*. Aquest valor indica el nombre de *features* que li permetem provar al model a cada arbre. En el cas de l'opció de *sqrt*, s'escull com a valor la arrel quadrada del nombre de *features* del *dataset*. Generalment l'increment de *max_features* implica un

increment en el rendiment, però això no és sempre cert ja que això decrementa la diversitat dels arbres.

Per *max_depth* tenim un valor de 25. Com indica el nom, aquest valor canvia la profunditat dels arbres del *Random Forest*

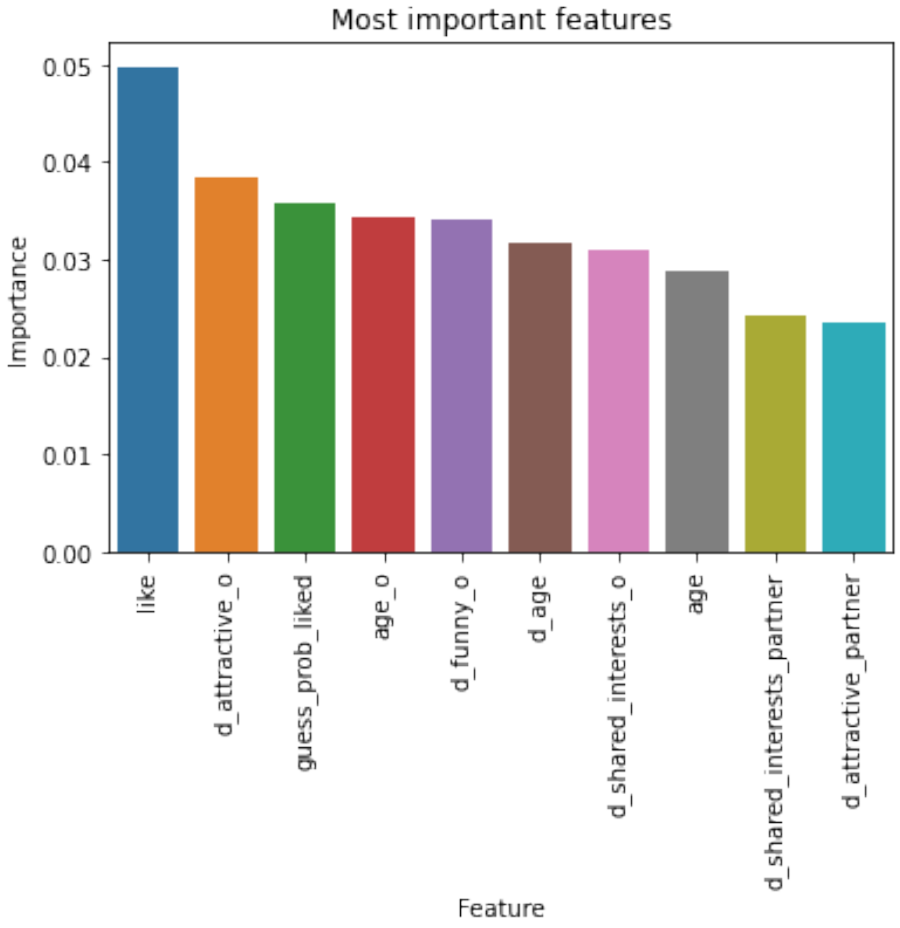


Figure 5: Les 10 *features* més rellevants de Random Forest

El cas del *Random Forest*, no tenim els coeficients com en el cas del models lineals de l'apartat anterior, però a canvi, ens proporciona directament un valor numèric amb la importància de cada *feature*. Veiem que la importancia que ens ha extret el *Random Forest* no és molt diferent a la que ja havíem vist als altres models. La variable de *like* segueix encapçalant la llista i, com ja hem comentat, això no ens sorprèn. Com ja havíem vist en els altres models, les *features* que tenen informació sobre l'edat dels participants són significativament importants a l'hora de decidir si hi ha un *match*.

7 Final Choice

Per a la tria del model final hem decidit que depèn de l'objectiu que vulguem aconseguir en triaríem un o un altre.

En primer lloc, en cas de que el nostre objectiu fos aconseguir un model que fes la predicció de si dues persones son compatibles per fer *match* amb la major precisió, escolliríem la *Linear Regression* ja que és el model que ens ha donat millors resultats a l'hora de predir. A partir de la *cross-validation* que hem dut a terme amb els models, hem extret la mitjana de la *accuracy* i no hem observat un canvi significatiu amb la que ens ha donat finalment el model. Això ens indica que el model és comporta de manera correcta amb dades que encara no ha vist.

En segon lloc, si el nostre objectiu fos fer un estudi a partir dels resultats del model, ens interessa molt el atribut de *feature importance* que ofereix el *RandomForestClassifier* que hem usat, ja que ens indica de manera numèrica quines són les *features* més importants de manera directa, i per tant, d'aquesta manera podem saber quins son els elements més importants en *speed dating*, i això ens pot portar a extreure conclusions basades en la importància de cada tret de les persones. Igual que amb els altres models, la precisió obtinguda fent la *cross-validation* no ha variat significativament respecte a la del model final, així que podem afirmar que el model és robust davant noves dades.

8 Report

Tot i que en un primer inici, el *dataset* semblava molt bo en quant a la varietat de tipus de *features*, quantitat d'aquestes i quantitat de mostres. La veritat és que, quan vam intentar quedar-nos amb les col·lumnes numèriques de les valoracions personals dels participants i parelles (explicat a la secció 3.2), ens vam adonar que, un cop eliminades les variables amb molts *NaNs*, les col·lumnes restants eren totes *numèriques*. Això, suposem, no seria un problema si de debò estiguéssim fent un estudi del *dataset* estadístic però, com que el nostre objectiu és aprendre, vam decidir escollir les variables on s'indicava la mateixa informació però de manera categòrica.

En quant a la representació visual dels models, no sabíem com interpretar els resultats d'aquells que són no lineals, hem llegit informació i mirat codis d'exemple que podem trobar a Internet i cap d'ells funcionava per les dades que tenim. Això, ens ha generat molt d'estrès i de frustració al llarg de la pràctica.

Estem molts satisfets amb l'*accuracy* que assoleixen tots les nostres models i creiem que la tassa d'error és acceptable. També hem pogut extreure conclusions interessants i divertides amb els diferents models.

Aquest mateix estudi es pot extendre afegint-hi més variables a tenir en compte o enfocant aquestes d'una manera diferent o més específica com fa l'estudi que

hem comentat a la secció 2. Això sí, s'ha de tenir en compte que estem vivint una època en la que les mentalitats de les diferents societats estan canviant constantment. Estem vivint un moment on es lluita pels drets de totes les persones independentment de la orientació sexual, de gènere, de l'origen, la capacitat econòmica, etc. Així que, vist des d'aquest punt de vista, el nostre *dataset* es troba molt obsolet i limitat. Creiem que les conclusions que s'obtinguin de treballar amb ell s'hauria de posar en context amb les característiques de la societat de la època originària del lloc on s'han recollit les dades.

En general, creiem que hem fet un bon estudi, ens hem informat i après molt durant aquests dos mesos, ha estat una pràctica divertida alhora que frustrant però així són els mons desconeguts quan ens introduïm en ells per primer cop.

References

[1] Ray Fisman and Sheena Iyengar. *SpeedDating*. 2016. URL: https://www.openml.org/d/40536.

[2] Keith McNulty. *speed_dating*. https://github.com/keithmcnulty/speed_dating. 2020.

[3] Keith McNulty. *What Matters in Speed Dating?* 2020. URL: https://towardsdatascience.com/what-matters-in-speed-dating-34d29102f6cb.

[4] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.