eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
INFORMATIKA FAKULTATEA
FACULTAD DE INFORMÁTICA

Informatika Ingeniaritzako Gradua

Konputazioa

Gradu Amaierako Lana

Egoera ez-normalen detekzioa dronaren motorrean

Egilea

Isabel Losantos Nekotxea

2019

eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
INFORMATIKA FAKULTATEA
FACULTAD DE INFORMÁTICA

Informatika Ingeniaritzako Gradua

Konputazioa

Gradu Amaierako Lana

Egoera ez-normalen detekzioa dronaren motorrean

Egilea

Isabel Losantos Nekotxea

Zuzendaria(k)

Itziar Irigoien Garbizu

Laburpena

Aireontziaren segurtasuna handitzeko, anomalien detekziorako gaitasuna handitzea garrantzitsua da. Gradu amaierako lan honetan, dronaren hainbat sentsore ezberdinetatik lortutako datuen analisia oinarri harturik, anomalien detekziorako aproposak diren bideak aztertzea eta egokienak direnak hautatzea izan da lan honen helburu nagusia. Helburu hori lortzeko, datuen analisia ahalbidetzen duten teknika ezberdinak eta ikasketa automatikoaren algoritmo batzuk erabili dira.

Gaien aurkibidea

Irudien aurkibidea

Taulen aurkibidea

1. KAPITULUA

Sarrera

Aeronautika sektorea iraultza handi baten barruan murgilduta dago, sektoreak berak dituen eskakizunen ondorioz. Eskakizun horiek isuria gutxitzea (CO_2, NO_x eta zarata) eta segurtasuna handitzea dira batik bat, eta gainera azken bi horiek etengabe handitzen ari den merkatuarekin (15 urtean behin bikoiztu egiten da) bateratuak egon behar dute.

Azken urteetan, Euskadi izan da Europako bigarren herrialdea ikerketan itzulkin ekonomiko handiena izan duena (*Clean Sky* [1] programa Europarrean). Horri esker, kontrol banatuetan oinarritutako teknologiek, euskal industria aeronautikan oihartzun handia edukitzeko aukera izango dute.

Gainera, sistema banatuak industria edo konputazio ingurumenez gain, beste hainbat arlo ezberdinetan erabiltzen hasi dira. Izan ere, aire-nabigazioa elektrifikatzeak, sistema aeronautikoen deszentralizazioaren erabilpena bultzatu du, aireontzien segurtasuna eta energia-eraginkortasuna handitzeko helburuarekin.

Sistema banatuetan, edozein nodoren erauzketak ez luke sareko beste edozein nodoren deskonexioa sortuko. Hau da, nodo guztiak lotuta daude bata bestearekin eta modu independenteki lan egiten dute, gune lokal batetik edo batzuetatik igaro beharrik izan gabe.

[1] Clean Sky: Europako ikerketa programa bat da, aireontziek sortzen dituzten soinu-maila, gas-isurketa eta CO_2 kopurua gutxitzeko teknologia berritzaileak erabiltzen dituena.

Honen helburua prozesadorea eta informazioaren memoria deszentralizatzea [2] [3] da. Honi esker, energia-eraginkortasun handiagoa, prozesadore eta abiadura banatu handiagoa eta segurtasun handiagoa lor daiteke.

Sare handien edo sistema banatuen arteko komunikazioaren bermatzea *Blockchain* [4] izeneko teknologia berritzaileari esker lor daiteke. Izan ere, gailuen arteko autentifikatzea, konfidentzialtasuna, uko egitea ekiditea edo datuen nahiz sistemen pribatutasuna bezalako arazoak konpontzeko aukera ematen du.

Hori dela eta, gaur egun aireontzietan erabiltzen ari diren arkitektura deszentralizatuen eta propultsio elektrikorako edo sare-elektriko banatuetarako planteatu diren sistema banatuen arteko trantsizioa erabateko erronka bat izango da kontrol banatu seguru eta eraginkorrerako.

Erronka hori dela medio, CODISAVA [5] partzuergoak kontrol banatuen adimen eta algoritmoen garapenean ardaztu da. Izan ere, garapen horiei esker, aireontzia osatzen duten sistema eta azpisistema ezberdinen energia-eraginkortasuna eta segurtasuna hobetzea espero da eta hori da CODISAVA-ren helburua. Gradu amaierako lan hau CODISAVA-ren helburuak ahalbidetzearekin loturik dago: datuen analisirako eta anomalien detekziorako tekniken aplikazioa sistema banatuetan.

Informazio hori kontuan izanik, aireontziaren segurtasuna handitzeko eta arriskuak ekiditeko metodo bat, dronaren motorrean dauden sentsoreek bildutako informazioa analizatzea izan daiteke.

Proiektu honen abiapuntua hain zuzen ere, dronaren sentsoreek bildutako fitxategiak atzitzea izan zen. Behin fitxategiak deskargatu eta bildu ondoren, fitxategiko datuak analizatu ziren metodo ezberdinak erabiliz, eta, fitxategi horietan zeuden datuak "normaletan"edo "anomaloetan"bereiztea ideia ona izan zen edo ez ondorioztatuko da.

[2] Sistema zentralizatuetan, nodoak ezin dira haien artean komunikatu, lotuta dauden nodo zentralizatuarekin bakarrik komunikatzen dira

[3] Sistema deszentralizatuetan, nodo erreguladoreren batek huts egiten badu, multzo horretako nodo batek baino gehiagok deskonektatzea eragin dezake

[4] Blockchain: Teknologia

[5] CODISAVA: Control distribuido avanzado para la seguridad y la eficiencia energética del transporte aéreo, Elkartek 2019 deialdiko ikerketa proiektua.

2. KAPITULUA

Proiektuaren Helburuen Dokumentua

Aurreko atalean lan honen jatorria azaldu ondoren, proiektu hau aurrera eramateko zehaztu ziren helburuak, eta, helburu horiek lortzeko garatu zen plangintza azalduko da atal honetan.

Dronaren egoera ez-normalen detekziorako baliogarriak izan zitezkeen prozedurak bilatzea izan zen proiektu honen helburu nagusia. Prozeduren bilaketa horietarako abiapuntua, motorrean zeuden hainbat sentsoreetatik jasotako informazioa izan zen.

Helburu hori lortzeko, beste hainbat helburu definitu ziren proiektuan zehar:

- Egoera normalen azterketa eta deskribapena egin egoera ez normalen detekziorako planteamendua egiteko.
- Egoera normalen eta ez normalen ezaugarri nagusiak eraiki, anomalien detekziorako teknikak aplikatu ahal izateko.
- Lortutako emaitzekin datu normalen eta ez-normalen arteko erlazioak ezagutu eta egoera anomaloak baldin bazeuden, detektatu.

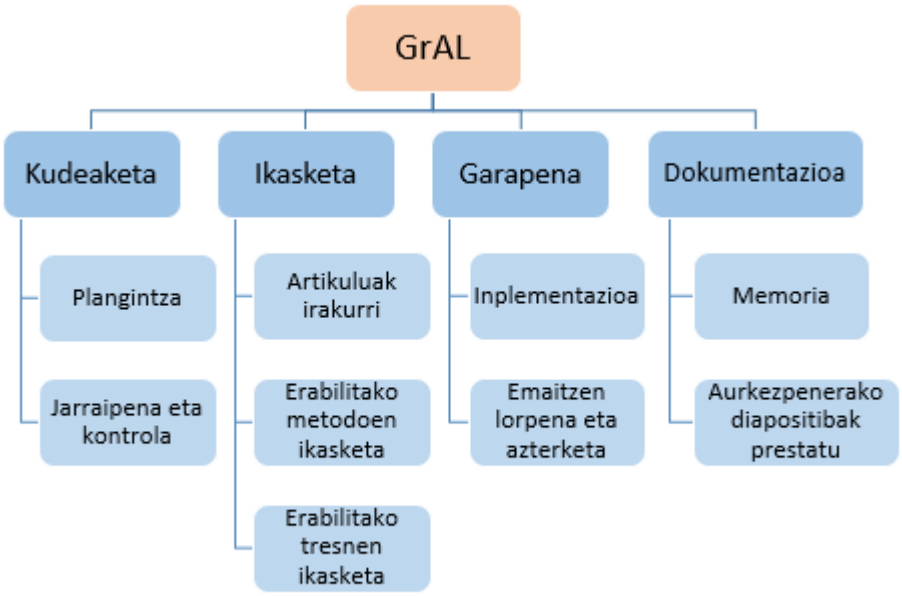
Helburu horiek lortzeko pausoak hobeto azalduko dira lan honetan zehar, bai Oinarri teorikoak atalean bai eta Garapena atalean ere (3. eta 5. atalak).

2.1 Proiektua garatzeko plangintza

Azpiatal honetan, proiektua garatzeko egin zen plangintza azalduko da. Alde batetik, plangintza hori aurrera eramateko sortu ziren atazak azalduko dira. Bestetik, ataza horiei eskaini zitzaien denbora taula baten bidez adieraziko da.

2.1.1 Lan-paketeen identifikazioa

Plangintzarako egin zen lan banaketa erakusteko, LDE diagrama erabili da. Hurrengo irudian edo LDE diagraman ikus daitekeen moduan, plangintza lau multzo ezberdinetan banatu zen. Hauetako multzo bakoitza, beste azpi-multzotan banatu ziren.



2.1 Irudia: LDE diagrama

Lan-pakete bakoitzaren azalpena:

- Kudeaketa: lan-pakete hau, **Plangintza** eta **Jarraipen eta kontrola** lan-paketeetan banatu zen. **Plangintza**, proiektuan garatu beharreko lan guztiaren deskribapena jasotzen zuen, uneoro zer egin behar zen adieraziz. **Jarraipen eta kontrola**, proiektua garaiz amaitzea bermatu zuen.

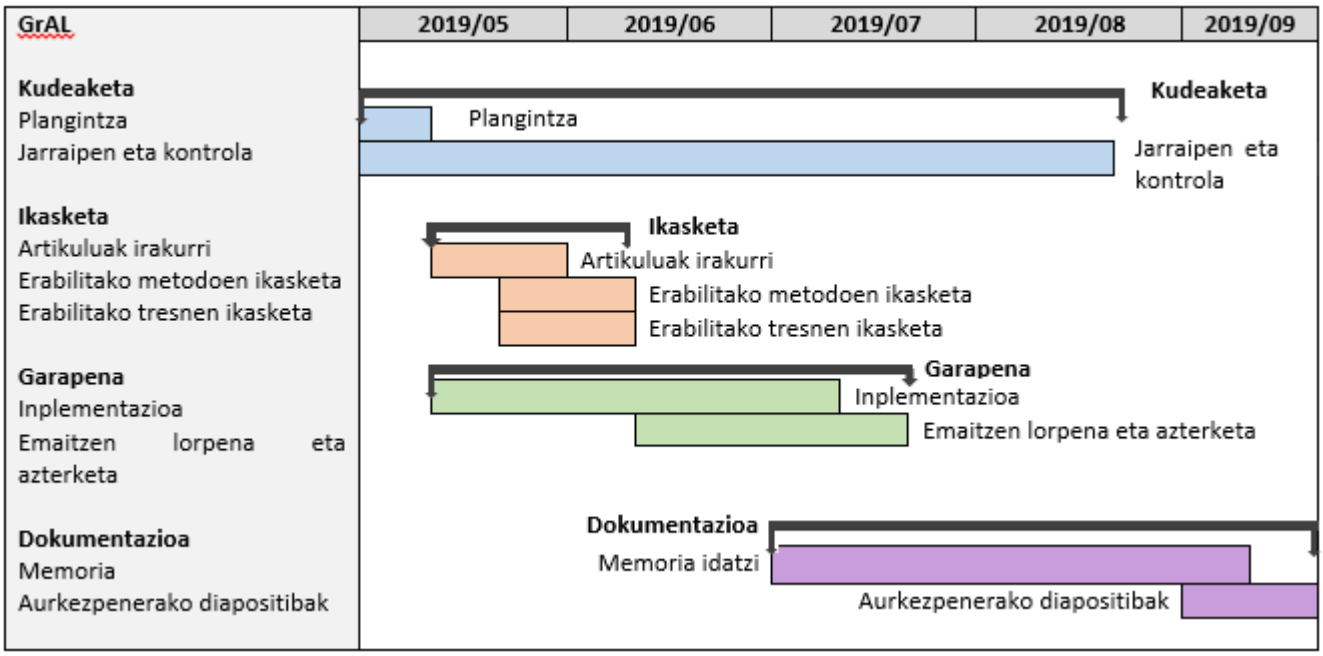
- Ikasketa: lan-pakete hau, hiru lan-pakete ezberdinetan banatu zen: **Artikuluak irakurri**, **Erabilitako metodoen ikasketa** eta **Erabilitako tresnen ikasketa**. Laburbilduz, lan-pakete honek proiektuan zehar garatu ziren tresnen zein metodoen oinarri teorikoen lana bildu zuen. Horretarako, hainbat artikulu irakurri ziren eta proiektua garatu zen ingurunean (R studio) probak egin ziren funtzionamendua ikasten joateko.

- Garapena: lan-pakete hau, bitan banatu zen: **Inplementazioa** eta **Emaitzen lorpena eta ebaluaketa**. Inplementazio atalean, sortu ziren funtzioak, prozedurak zein metodoak garatu ziren. **Emaitzen lorpena eta ebaluaketa** atalean, inplementazio atalean lortu ziren emaitzak aztertu ziren, emaitzen inguruan ondorioak atera ahal izateko.

- Dokumentazioa: proiektuaren entrega eta aurkezpenarekin lotuta zegoen. **Memoria** lan-paketeak, proiektuaren dokumentazio osoa jaso zuen. **Aurkezpenerako diapositibak prestatu** lan-paketeak, proiektua entregatu ondoren egin beharreko aurkezpenarekin loturik zegoen. Bertan, aurkezpena egin ahal izateko beharrezkoak izan ziren baliabideak bildu ziren.

2.1.2 Planifikatutako egutegia

Hurrengo Gantt diagramaren bitartez, ataza eta lan-pakete bakoitzari eskaini zitzaizkion denbora eta epe-mugen aurreikuspena adieraziko da. Orokorrean, lehen urratsak oinarri teorikoen ezagumena izan ziren, ondoren garapena eta azkenik dokumentazioa:



2.2 Irudia: Gantt diagrama

Proiektua ondo garatzeko, gutxienez 300 orduko dedikazioa eman behar zitzaion. Hori dela eta, ordu kopuru hori zatitu egin zen, lan-pakete bakoitzari dedikatuko zitzaion denbora aurreikusiz. Lan-paketeei denbora esleitzeko, proiektuan zehar egon zitezkeen aldaketak eta desbiderapenak kontuan hartu ziren. 2.1 taulan, lan-pakete bakoitzari dedikatu zitzaion denbora finala adieraziko da:

LAN-PAKETEA	*ORDU KOPURUA*
Kudeaketa	**35**
Plangintza	15
Jarraipen eta kontrola	20
Ikasketa	**60**
Artikuluak irakurri	20
Erabilitako metodoen ikasketa	20
Erabilitako tresnen ikasketa	20
Garapena	**110**
Inplementazioa	90
Emaitzen lorpena eta azterketa	20
Dokumentazioa	**100**
Memoria	80
Aurkezpenerako diapositibak	20
GUZTIRA	**305**

2.1 Taula: Lan-pakete bakoitzari dedikatu zitzaion ordu kopurua

2.2 Lan metodologia

Proiektuari, egunero hainbat ordu dedikatu zitzaizkion. Egunean lan egingo zen ordu kopuru zehatza jakitea ezinezkoa zen. Izan ere, egunerokotasunaren arabera, egun batzutan lan gehiago eta beste egun batzuk lan gutxiago egingo zelako.

Bestalde, proiektuak aurrera egiten zuen heinean, zalantzak sortu ziren. Zalantzak edo trabatzeak sortzen ziren bakoitzean, tutorearekin bilerak eduki ziren. Orokorrean, astean behin tutorearekin elkarretaratze bat egin zen.

Egunero, proiektuan aurreratzen zen gauza bakoitzaren kopia *Google Drivera* igotzen zen, lana galduko ez zela ziurtatzeko.

2.3 Emangarriak

Hauek dira sortu ziren emangarriak:

- Garapena: inplementazioa eta lortutako emaitzen azterketa, gutxienez memoria entregatu behar zen eguna baino hilabete eta erdi lehenago amaitzea espero zen.
- Memoria: 2019ko irailaren 8an entregatu behar zen. Data hori baino egun batzuk lehenago amaitzea espero zen, egon zitezkeen ustekabeak ekiditeko.
- Aurkezpenaren diapositibak: memoriaren defentsa edo aurkezpena, 2019ko irailaren 16-20an izan zen. Aurkezpena, irailaren 16a baino egun batzuk lehenago prest edukitzea espero zen.

2.4 Arriskuen kudeaketa

Proiektuan zehar egon zitezkeen arriskuak identifikatu eta hauei aurre egiteko modua azalduko da:

- Ezjakintasuna: proiektu edo lan berri bat egiten hasten garen bakoitzean, hasieran nahiko galduta ibiltzen gara eta batzutan aurrera egiteko oztopo bat izan daiteke.

Horren aurrean, Interneten ahalik eta informazio gehien bilatu zen gauzak ondo ulertzeko, eta tutoreari zalantzak lehenbailehen galdetu zitzaizkion.

- Informazio galera: proiektuan egindako aurrerapenak eta inplementatutako kodea ez galtzeko, egunero egiten zen lana trinkotzen zen eta *Google Drivera* horren kopia igotzen zen.
- Denbora falta: egunerokotasuna dela eta, ustekabeak sor daitezke, eta, horrek, proiektua garaiz ez amaitzea eragin dezake. Hori saihesteko, plangintza ahal izan zen neurrian jarraitu zen, lan guztia azken unerako utziko ez zela bermatuz.
- Komunikazioa: komunikazio arazoak edo gaizki-ulertuak ekiditeko, tutorearekin egin ziren bilerak aldez aurretik hitzartzen ziren.

2.5 Baliabideen kudeaketa

Proiektuaren garapenean, baliabide ezberdinak erabili ziren:

- Proiektuaren inplementazioa egiteko eta emaitzak lortzeko, *R* lengoaia eta *R studio* softwarea erabili ziren.
- Memoria idazteko aldiz, *Latex* editorea erabili zen *Overleaf* tresnarekin batera. Aurkezpenaren diapositibak egiteko, *Microsoft PowerPoint* erabili zen.
- Egunero egiten zen lanaren aurrerapenak gordetzeko, *Google Drive* erabili zen.

2.6 Proiektuaren hartzaileak edo interesatuak

Dudarik gabe, proiektu honen interesatu nagusiena proiektuaren garatzailea bera izan zen. Bera baitzen proiektua garaiz amaitzeko ardura nagusiena zuena.

Bestalde, proiektuaren zuzendaria zegoen, Itziar Irigoien. Ikaslea gidatzeaz gain, proiektua zuzendu, zalantzak argitu eta emaitzak arrakastatsuak izango zirela ziurtatu zuen.

Azkenik, proiektuaren defentsa egunean epaimahaian egon ziren irakasleak daude. Haien ardura proiektua ebaluatzea izan zen.

3. KAPITULUA

Oinarri teorikoak

Eranskin honetan, proiektuan zehar landu ziren oinarrizko kontzeptuak azalduko dira, dronaren motorrean egoera ez-normalen detekzioa ahalbidetu zuten datu-analisiko tekniken oinarri teorikoa ezagutzeko.

3.1 Distantziak

Distantzia kontzeptua bere propietateekin batera, oso erreminta garrantzitsua da datuen analisirako. Batetik, distantzia bati esker, hipotesien arteko kontrasteak ikusi, parametroak konparatu, estimatzaileen propietate asintotikoak ikasi ... daitezke. Eta bestetik, distantziari esker, ulertzeko errazak diren irudikapen geometrikoak lortu daitezke, ikertzaileari datuen arteko egitura ulertzea errazten diotenak.

Objektuen edo elementuen arteko distantziaren kontzeptuak, geometrikoki aldaera anitzeko analisiaren teknika ugari interpretatzea ahalbidetzen du. Eta objektu horiek espazio metriko egoki baten puntuak bezala errepresentatu daitezke.

Orokorrean, i eta j bi elementuen arteko distantzia edo diferentzia neurri bat da. Neurri hori $d(i,j)$ espresioarekin adierazten da. Neurri horrek bi elementuen arteko antzekotasuna, edo hobeto esanda, ezberdintasuna neurtzen du ezaugarri kuantitatibo edo kualitatiboekiko. $d(i,j)$ zenbat eta handiagoa izan, orduan eta handiagoa izango da i eta j elementuen arteko diferentzia.

Matematikan, matrize baten elementuek puntuen arteko distantzia adierazten dutenean distantzia matrizea deritzo. Puntu horiek binaka hartzen dira. n, datu-multzo bateko laginaren tamaina izan ohi da. Baina, Ω, n elementuez osatutako datu-multzo finitu bat ere izan daiteke. Ω datu-multzoaren i, j gizabanako edo elementuen $d_{ij} = d(i, j)$ distantzia, neurri simetriko ez negatibo bat da, haien arteko diferentzia kuantifikatzen duena aldagaiekiko. d, distantzia matrize baten bidez adierazi daiteke:

$$D = \begin{pmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nn} \end{pmatrix} \tag{3.1}$$

non $d_{ii} = 0, d_{ij} = d_{ji}$ den.

Bi aldagai zenbateraino antzekoak edo ezberdinak diren neurtzeko, dibergentzia edo ezberdintasun indizea erabiltzen da. Indize anitz dago, eta bakoitzak bere propietate eta erabilpenak dauzka. Indize horiek, distantzian oinarritutako adierazleak edo korrelazio koefizienteetan oinarritutako adierazleak izango dira.

Nahiz eta distantzian oinarritutako hainbat adierazle ezberdin dauden, lan honetan erabili ziren distantziak, distantzia euklidearra eta Mahalanobis distantzia izan ziren. Bestalde, aldagaien arteko diferentziak ikusteko, korrelazio-distantzia erabili zen. Hori dela eta, kontzeptu horiek banan-banan azalduko dira.

3.1.1 Distantzia euklidearra

Espazio euklidear bateko bi puntuen arteko distantzia "arrunta"da. Hiru edo dimentsio gehiagoko espazioko puntuen arteko distantziak kalkulatzeko ere balio du. Eta ez horretarako bakarrik, plano edo zuzen bateko bi puntuetatik definitutako segmentu baten luzera kalkulatzeko ere. Bere oinarria Pitagorasen teoremaren bitartez ondorioztatzen da.

Modu orokorrean, Ω datu-multzoko n elementuek $X(n \times p)$ matrizea osatzen dute, non p aldagai estatistikoen (kuantitatiboak) kopurua den. Datu-multzo horren distantzia euklidearra horrela definitzen da:

$$d(x_i,x_j) = \sqrt{\sum_{z=1}^{p}(x_{iz}-x_{jz})^2} = \sqrt{(x_{i1}-x_{j1})^2+(x_{i2}-x_{j2})^2+...+(x_{ip}-x_{jp})^2} \quad (3.2)$$

$\forall i,j \in \{1,...,n\}$. Izan ere, i eta j, $n \times p$ dimentsioko matrizearen lerroak adierazten dituzte.

Distantzia edo metrika euklidearrak, baldintza nahikoak betetzen ditu metrikatzat hartzeko, horregatik propietate hauek betetzen ditu:

1. $d(x_i,x_j) \geq 0$
2. $d(x_i,x_j) = d(x_j,x_1)$ *Propietate simetrikoa*
3. $d(x_i,x_j) = 0$
4. $d(x_i,x_j) \leq d(x_i,x_t) + d(x_t,x_j)$, $\forall i,j,t \in (1,...,n)$ *Desberdintza triangeluaren propietatea*
5. Baldin $d(x_i,x_j) = 0$, orduan, $x_i = x_j$

Hala ere, distantzia euklidearrak bi arazo dauzka:

- Distantzia sentikorra da aldagaien unitateen neurriarekiko. Hau da, balio altuak dituzten aldagaien unitateen arteko diferentziak eragin handiagoa izango du balio baxuagoak dituzten aldagaien unitateen diferentziak baino. Beraz, eskala aldaketak elementuen arteko distantzia mugatzen du. Horren soluzio posible bat aldagaien aurretiko sailkapen bat edo distantzia euklidear normalizatua izango litzateke.
- Aldagaien izaera batzutan arazo bat izan daiteke. Erabilitako aldagaiak korrelazioan jarrita badaude, aldagai hauek emango diguten informazioa erredundantea izango da. Aldagai batzuen arteko diferentziak beste aldagai batzuen diferentziengatik azaldu ahalko lirateke. Eta horren eraginez, distantzia euklidearrak elementuen arteko diferentzia edo dibergentzia handituko luke. Honen soluzio posible bat, jatorrizko aldagaiak aztertu beharrean, osagai nagusiak aztertzea da. Beste soluzio posible bat, Mahalanobis distantzia erabiltzea izango litzateke.

Orokorrean, aldagaiak homogeneoak eta antzeko unitateetan neurtuta daudenean edo bariantza matrizea ezezaguna den kasuetan distantzia euklidearra erabiltzea komenigarria da.

3.1.2 Mahalanobis distantzia

Distantzia normalizatu bat da, desbideratze unitateen bidez adierazten dena. Aldagaien arteko korrelazioak kontuan hartzen ditu, hau da, aldagaien arteko erredundantzia. Distantzia anitzeko zorizko bi aldagaien antzekotasuna determinatzen du. Zorizko aldagaien korrelazioa kontuan hartzeagatik distantzia euklidearratik bereizten da.

S, $n \times p$ dimentsioko X matrizearen bariantza-kobariantza matrizea izanik eta x_i eta x_j, X matrizeko bi lerro izanik, Mahalanobis distantzia horrela definitzen da:

$$d_m(x_i,x_j) = \sqrt{(x_i - x_j)^T S^{-1}(x_i - x_j)} \tag{3.3}$$

non S^{-1}, S bariantza-kobariantza matrizearen alderantzizkoa den.

Mahalanobis distantziak, propietate hauek betetzen ditu:

1. Erdipositiboa: $d(x_i,x_j) \geq 0$ eta $d(x_i,x_j) = 0$ baldin $x_i = x_j$
   Koordenatu berdineko bi puntuen arteko distantzia zero da, eta koordenatu desberdinak badituzte, distantzia positiboa da, negatiboa inoiz ez.

2. Simetria: $d(x_i,x_j) = d(x_j,x_i)$
   Intuitiboki, x_i eta x_j-ren arteko distantzia x_j eta x_i-ren arteko distantziaren berdina da.

3. Desberdintza triangeluarra: $d(x_i,x_j) \leq d(x_i,x_t) + d(x_t,x_j), \forall x_i,x_j,x_t \in X$

Mahalanobis distantzia, distantzia euklidearrak zituen bi arazoak konpontzeko gai da. Batetik, eskala aldaketekiko inbariantea da, eta hortaz, ez dago unitateen neurrien menpe. Bestetik, S matrizea erabiltzen duenez, aldagaien arteko korrelazioak kontuan hartzen dira, eta horri esker erredundantziaren efektua zuzentzen da.

3.1.3 Korrelazio distantzia

Korrelazio kontzeptua azaltzea ezinbestekoa zen lan honetan. Izan ere, aldagaien arteko distantziak neurtzerakoan eta dimentsio anitzeko mailaketa erabiltzean, aldagaien arteko korrelazioa erabili zen kalkuluetarako.

Bi elementu edo gehiagoren arteko loturari korrelazioa deritzo. Testuinguruaren arabera, kontzeptu hau modu ezberdinetan erabili daiteke.

Matematika eta estatistika esparruetan, korrelazioak aldagai ezberdinen arteko proportzionaltasunari eta erlazio linealari egiten dio erreferentzia. Aldagai baten balioak sistematikoki beste aldagai baten balioekiko aldatzen badira, bi aldagai horiek korrelazioan jarrita daudela esaten da. Adibidez, X_1 eta X_2 aldagaiak baditugu, X_1en balioak handitzen baditugu, X_2ren balioak handitzen dira. Eta, era berean, X_2ren balioak handitzen baditugu, X_1en balioak areagotzen dira. Beraz, X_1 eta X_2 aldagaien artean korrelazio bat existitzen da. Korrelazioa, aldagai ezberdinen mendekotasunetik erregistratzen den neurria dela esan daiteke.

$n \times p$ dimentsioko X matrizearen R korrelazio matrizea, $p \times p$ dimentsiotako matrize karratu bat da, aldagai bikotekide bakoitzaren korrelazio koefizienteez osatua dagoena. Bere diagonal nagusia bat zenbakiez osatua dago, eta diagonal nagusian ez dauden (i, j) elementuetan haiei dagozkien r_{ij} korrelazio koefizientea dago.

$$R = \begin{pmatrix} 1 & r_{12} & \cdots & r_{1p} \\ r_{21} & 1 & \cdots & r_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ r_{p1} & r_{p2} & \cdots & 1 \end{pmatrix} \tag{3.4}$$

Korrelazio matrizea simetrikoa eta positiboa izatearen propietateak mantentzen ditu. Gainera, bere determinantea ez negatiboa da (beti bat edo bat baino txikiagoa izango da).

Korrelazio matrizea sortu ondoren, korrelazio koefiziente horiek erabiliko dira distantzia-matrizea sortzeko, non distantzia-matrizeko koefiziente bakoitza modu honetan kalkulatuko den:

$$d_{ij} = \sqrt{1 - r_{ij}}$$

Hori dela eta, distantzia-matrizea modu honetan osatuko da:

$$D = \begin{pmatrix} 0 & \sqrt{1-r_{12}} & \cdots & \sqrt{1-r_{1p}} \\ \sqrt{1-r_{21}} & 0 & \cdots & \sqrt{1-r_{2p}} \\ \vdots & \vdots & \ddots & \vdots \\ \sqrt{1-r_{p1}} & \sqrt{1-r_{p2}} & \cdots & 0 \end{pmatrix} \tag{3.5}$$

Aldagaien artean zenbat eta korrelazio handiagoa egon, orduan eta distantzia txikiagoa egongo da haien artean.

Lan honetan,korrelazio koefizienteak kalkulatzeko erabili zen korrelazioa, Pearsonen korrelazio koefizientea izan zen:

Pearsonen korrelazio koefizientea

Zorizko bi aldagai kuantitatiboen arteko neurri lineala da. Neurri honek, bi aldagaien arteko erlazioak zenbateko intentsitatea eta zer norabide duen zehazten du. Hau da, linealki erlazionatuta dauden bi aldagai ezberdinen kobariantza gradua neurtzen du. Pearsonen korrelazioa independentea da aldagaien eskalaren neurriarekiko.

Ω datu-multzoaren n elementuek $X(n \times p)$ matrizea osatzen dute. X_1 eta X_2 datu-multzo horren zorizko bi aldagai direla suposatuz, Pearsonen korrelazio koefizientea kalkulatzeko, formula hau aplikatu behar da:

$$r_{X_1X_2} = \frac{\sum X_{i1}X_{i2} - n\bar{X_1}\bar{X_2}}{(n-1)s_{X_1}s_{X_2}} = \frac{n\sum X_{i1}X_{i2} - \sum X_{i1}\sum X_{i2}}{\sqrt{n\sum X_{i1}^2 - (\sum X_{i1})^2}\sqrt{n\sum X_{i2}^2 - (\sum X_{i2})^2}} \tag{3.6}$$

Korrelazio indizearen balioa $[-1,1]$ tarteko balioak hartzen ditu, eta balio horrek aldagaien arteko erlazioa adierazten du:

- $r = 1$ bada, korrelazio positibo hobezina existitzen da. Indize horrek, aldagaien artean erabateko mendekotasuna dagoela adierazten du, hau da, *zuzeneko mendekotasuna* daukate. Aldagaietako bat handitzen bada, bestea ere handitzen da proportzionalki.
- $0 < r < 1$ bada, korrelazio positiboa existitzen da.
- $r = 0$ bada, aldagaien artean ez da erlazio linealik existitzen.
- $-1 < r < 0$ bada, korrelazio negatiboa existitzen da.
- $r = -1$ bada, korrelazio negatibo hobezina existitzen da. Indize horrek, aldagaien artean erabateko mendekotasuna dagoela adierazten du. Daukaten mendekotasunari *alderantzizko erlazioa* deritzo. Aldagaietako bat handitzen denean, bestea proportzionalki txikitzen da.

3.2 Dimentsio anitzeko mailaketa: Koordenatu nagusiak

Dimentsio anitzeko mailaketaren jatorria Queteleten gizarte-zientziak kuantifikatzeko esfortzuetaraino atzera egin dezake. Hala ere, bere jaiotza 50 hamarkadan egin ziren psikologia esperimentalaren ikerketetan dago. Ikerketa horiekin gizabanako ezberdinei aplikatutako estimuluen arteko antzekotasunak aurkitu nahi ziren. Hala ere, bere gaur egungo garapena Torgerson, Shepard, Kruskal eta Gower zientifikoei esker egindako ikerketei esker da. Existitzen diren metodoak bitan sailkatzen dira. Batetik, metrikoak, hasierako matrizea distantziez osatua dagoenean. Bestetik, ez-metrikoa, hasierako matrizea antzekotasunez osatua dagoenean.

Dimentsio anitzeko mailaketa edo koordenatu nagusien analisia, aldagai anitzeko analisi teknika da, datuak aztertzeko eta datuen arteko antzekotasunak irudikatzeko baliogarria dena. Azterketa hau, multzo baten n elementuen arteko distantzia edo antzekotasunak adierazten dituen $n \times n$ matrize karratu batetik hasten da, D. Matrize hori aldagai ortogonalez, $y_1, ..., y_p$, osatutako multzo baten bidez irudikatzea da helburua. Aldagai ortogonal horiei *koordenatu nagusiak* esaten zaie, non $p < n$ den eta aldagai berri hauekiko elementuen arteko distantzia euklidearrak jatorrizko matrizearen distantziarekiko ahalik eta berdinena den. Hau da, hasierako D matrizetik, $n \times p$ dimentsiotako Y matrizea lortu nahi da (p, n elementu bakoitzak dituen aldagaiak dira), non elementuen arteko distantzia euklidearrak hasierako D matrizea erreproduzituko duten. Orokorrean, oso zaila da hasierako distantzia berdinak irudikatuko dituzten p aldagaiak aurkitzea p oso txikia bada, baina distantzia horretara parekatzen diren aldagaiak aurkitzea posiblea da.

Hitz gutxitan, dimentsio anitzeko mailaketaren helburua elementuak deskribatzea eta interpretatzea da. Elementu ugari existitzen badira, antzekotasun-matrizea oso handia izango da eta matrize horren irudikapenak, elementuen bitartez, bere egitura ulertzen lagunduko digu: zein elementuk antzeko propietateak dituzten, elementuen artean taldeak agertzen diren edo ez, ohi kanpoko elementuak dauden edo ez etab.

Beraz, dimentsio anitzeko mailaketak elementuen arteko gertutasunetik edo distantziatik abiatuta, elementuetan ezkutaturik dauden egiturak aurkitzeko hainbat tekniken berri ematen die ikertzaileei.

Irteera, espazio-irudikapen bat da, puntuen konfigurazio geometriko bat alegia, mapa ba-

tean bezala. Konfigurazio bakoitzaren puntua elementu bakoitzari dagokio. Konfigurazio honek datuen egitura ezkutua islatzen du eta elementuen arteko ulermena eta ezagumena errazten ditu.

Koordenatu nagusiak nola kalkulatzen diren azaltzea ezinbestekoa da. Izan ere, horiek baitira hasierako D distantzia matrizean agertzen diren aldagaien arteko distantziak irudikatzen saiatuko diren aldagai ortogonalak.

3.2.1 Koordenatu nagusiak

$n \times n$ dimentsioko D matrizea daukagu, n elementuen arteko distantzia edo ezberdintasunak neurtzen dituena. Matrize horretatik, aldagaien arteko parekotasuna neurtzen duen Q matrizea eraikitzen da. Q matrize horren elementu bakoitza formula honen bidez kalkulatzen da:

$$q_{ij} = -\frac{1}{2}(d_{ij}^2 - d_{.j}^2 - d_{i.}^2 + d_{..}^2) \tag{3.7}$$

non $d_{.j}$ j.zutabeko elementuen batura den, $d_{i.}$ i.lerroko elementuen batura den, $d_{.j} = \sum_{i=1}^{n} d_{ij}$ den eta $d_{i.} = \sum_{j=1}^{n} d_{ij}$ den.

Ondoren, V eta Λ matrizeak kalkulatzen dira. Λ, $p \times p$ dimentsioko matrize diagonal bat da, Q matrizearen balio propio [1] ez-nuloez osatua dagoena. V, $n \times p$ dimentsioko matrize bat da, eta bere zutabeetan Q matrizeko balio propio ez-nuloei dagozkien bektore propioak ditu. Hortik abiatuz, Y matrizea sortzen da, eta bere zutabeak koordenatu nagusi deiturikoez osatua dago:

$$Y = (V\Lambda^{\frac{1}{2}}) \tag{3.8}$$

Koordenatu nagusiak kalkulatu ahal izateko, Q matrizearen balio propioak ez negatiboak izatea ezinbestekoa da 3.8 ekuazioa kontuan hartuz. Q matrizeari esker, distantzia matrizea metrika euklidearrarekin noiz den bateragarria egiaztatu daiteke. Hain zuzen ere, D distantzia matrizea metrika euklidearrarekin bateragarria dela esango dugu D erabiliz sortzen den antzekotasun matrizeari esker,

$$Q = -\frac{1}{2}PD^{(2)}P \tag{3.9}$$

[1] $Q \in \mathbb{R}^{n\times n}$ izanik, $\Lambda_{ii} \in \mathbb{R}^{p\times p}$ Q matrizearen balio propio bat izango da baldin eta soilik baldin $V_i \in \mathbb{R}^{n\times p}$ bektore ez nuloa existitzen bada, non $Q \cdot V_i = \Lambda_{ii} \cdot V_i$, $i = 1, \ldots, p$ eta $V_i \neq 0v$. V_i, Λ_{ii}-ren bektore propioa izango da

non $P = I - \frac{1}{n}11^t$ den. Proposamen horren arabera, D matrizearen kopia bat egin dezakeen metrika euklidear bat aurkitu daiteke. $D^{(2)} = (d_{ij}^2)_{ij}$ adierazi nahi du.

3.2.2 Koordenatu nagusien eraikuntza

$D^{(2)}$ distantzia karratuen matrizetik abiatuz, hurrengo pausoak egin behar dira:

- Q antzekotasun matrizea eraiki formula honen bitartez

$$Q = -\frac{1}{2}PD^{(2)}P$$

edo bere koefizienteen kalkulua eginez

$$q_{ij} = -\frac{1}{2}(d_{ij}^2 - d_{\cdot j}^2 - d_{i\cdot}^2 + d_{\cdot\cdot}^2)$$

- Q matrizearen balio propioak kalkulatu
- r balio propio handienak aukeratu, gainontzeko $n-r$ balioak zerotik hurbil egon daitezen. 0 beti izango da Q matrizearen balio propio bat, $P1 = 0$ delako, eta $Q1 = 0$, eta horren eraginez 0 Q matrizearen balio propio bat da 1 bektore propioarekin erlazionaturik dagoena.
- Koordenatu nagusiak kalkulatu, non V_r matrizearen zutabeak, Λ_r matrizean definitutako r balio propioei dagozkien r bektore propioez osatuak dauden. Modu horren arabera, lehen r koordenatu nagusiak horrela lortzen dira:

$$Y_r = V_r\Lambda_r^{1/2} \tag{3.10}$$

- Dagoeneko eraikita ditugun koordenatu nagusiak erabiliz, $n+1$ elementu berri bat, x_o, proiektatu nahiko balitz, hurrengo formula erabiliko da:

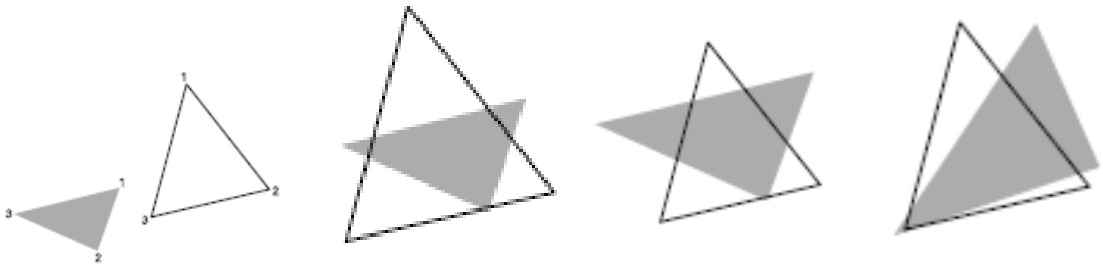
$$x_o = \frac{1}{2}\Lambda^{-1}Y'(b - d^{(2)})$$

non:

- Λ: Matrize diagonal bat da. Entrenamenduan parte hartu duten datuekin distantzia anitzeko mailaketa aplikatzeko erabiltzen den Q matrizearen balio propioez osatzen dena. Λ^{-1}, Λ matrizearen alderantzizkoa da.
- Y: Koordenatu nagusiez osatutako matrizea da. $Y^{'}$, Y matrizearen iraulia da.
- b: Q matrizearen diagonala da
- $d^{(2)}$: Distantzia-bektorea da. x_o elementuak gainontzeko n elementuekiko daukan distantziak biltzen ditu. $d^{(2)} = ((d(x_1,x_o))^2,(d(x_2,x_o))^2,...,(d(x_n,x_o))^2)$ adierazi nahi du.

3.3 Procrustes analisia

Procrustesen analisia (PA), datu-multzo baten bi konfigurazio konparatzeko balio duen matematikako erreminta da. Konfigurazio horiek entitate bereko bi aldaeretatik eratorriak dira. PAren helburua, n puntuen irudikapen ezberdinek n puntuen arteko barne erlazio ezberdinak erakusten dituzten zehaztea da.



(a) Jatorrizko koordenatuak **(b)** Zentratutako koordenatuak **(c)** Zentratutako eta mailakatutako koordenatuak **(d)** Zentratutako, mailakatutako eta biratutako koordenatuak

3.1 Irudia: Triangeluak erabiliz, Procrustesen analisia burutzeko egin beharreko hiru pausuen irudikapen grafikoa (Procrustesen gainazarpen metodoa erabiliz).

PA erabat aproposa da konfigurazioen bitartez irudikatutako bi egituren itxuren desberdintasunak aztertzeko. Metodo ezberdinak existitzen dira horretarako. Bat, karratu minimoen bitarteko PA da, Procrustes gainezarpen klasikoa izenarekin ezaguna dena. Bestea, erdibitzaile errepikapenaz egindako PA sendoa da. Procrustesen gainezarpen klasikoa, puntu

bikoteen diferentzien karratuen batura minimizatzea da. Geometrikoki, minimizazio hori puntuen arteko doikuntza ezaren batezbestekoa egiten lortzen da.

$p = 2$ dimentsioko n puntuen konfigurazioak nola konparatzen diren ikusteko, matematikako hainbat kontzeptu azalduko dira. Aipatu beharra dago, Procrustesen gainezarpen klasiko metodoarekin zerikusia dauzkaten matematikako oinarriak azalduko dira. Metodo hori baita lan hau aurrera eramateko erabili zena.

Procrustesen gainezarpen klasikoa $n \times 2$ dimentsiotako X eta Y matrizeak izango ditugu. Matrize hauen ilarak $p = 2$ dimentsioko n puntuen koordenatu kartesiarrekin bat etortzen dira.

$$X = \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ \vdots & \vdots \\ x_{n1} & x_{n2} \end{pmatrix} Y = \begin{pmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \\ \vdots & \vdots \\ y_{n1} & y_{n2} \end{pmatrix} \tag{3.11}$$

Bi matrizeek zentroidean daukate jatorria. Hori lortzeko, hauetako matrize bakoitza $I - P$ matrizearekin biderkatu dira. Non I $n \times n$ dimentsioko identitate matrizea den eta P $\frac{1}{n}$ koefizienteez osatutako $n \times n$ dimentsioko den.

$$I = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} P = \begin{pmatrix} \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \end{pmatrix} \tag{3.12}$$

$t = [t_1 t_2]$ translazio bektorea, $z > 0$ eskala faktorea eta R (matrize ortogonala) biraketa matrizea aurkitu behar dira. Batetik, konfigurazio bateko puntuen diferentzia karratuen batura minimizatzeko. Eta bestetik, gainontzeko konfigurazio horren bertsio biratua, eskalatua eta translazioa lortzeko.

$$min_{(t,z,R)} \sum_{i=1}^{n} = [(z[x_{i1}x_{i2}]R + t) - [y_{i1}y_{i2}]]^2 = min_{(t,z,R)} tr([(zXR + 1t) - Y]^{'}[(zXR + 1t) - Y] \tag{3.13}$$

tr trazak adierazten ditu. 1 zutabe-bektorea da n batekoz osatua dagoena. $X^{'}$, X matrizearen iraulia da.

Goiko formula minimizatu daiteke. Izan ere, $UBV^{'} = Y^{'}X$ da, non U, V $p \times p$ neurriko matrize ortogonalak diren eta $B = diag(b_{11}, b_{22}, ..., b_{pp})$ $p \times p$ neurriko matrize diago-

nal bat den, bere elementuak positiboak eta bere elementuak geroz eta txikiagoak diren ($B_{11} \geq B_{22} \geq ... \geq B_{pp} \geq 0$). Beraz, aurreko adierazpena minimizatzen duten t, z eta R balioak honako hauek dira:

$$t = \begin{bmatrix} 0 & 0 \end{bmatrix}, z = \frac{tr(B)}{tr(X'X)}, R = VU'$$

Beraz, balio berri horiek (3.13) formulan aplikatuz eta X matrizeari transformazio horiek aplikatuz, Procrustesen gainezarpen klasikoa lortzen da Y matrizearekiko. X matrizearen puntuen koordenatuen transformazioak desbiderazio karratuen batura minimoa daukate Y matrizearen koordenatuekiko. Minimo horri, Procrustes estatistikoa esango diogu.

X matrizearen translazioa, mailaketa eta biraketa eginez, Y matrizera ahalik eta gehien hurbiltzea da helburua. Horrela, prozesuaren amaieran lortzen dugun Procrustes estatistiko koefizienteari esker, X matrizea zenbateraino den Y matrizearen antzekoa ikusiko da.

3.4 Ikasketa ez-gainbegiratua edo clustering

Ikasketa automatikoaren algoritmo mota bat da. Datu-multzo bat talde homogeneoetan sailkatzeko erabiltzen da. Hasierako datu-multzoaren elementuak ez daude etiketatuak, eta, etiketa horiek sortzea ahalbidetzen duten elkartze teknikak erabili behar dira. Elkartze sistema honen helburua, objektuak multzo ezberdinetan sailkatzea da. Multzo berean elkartutako elementuak beraien artean oso berdinak eta beste multzoko elementuekiko desberdinak izango dira.

Orokorrean, hau da ebatzi behar den problema: $n \times p$ elementuez osatutako datu-multzoa emanik, elementuak $(X_1, ..., X_p)$ aldagaien informazioaz bereizten direnak, gure erronka elementu horiek klasifikatzeko gai izatea da. Horrela, talde (edo *cluster*) bereko elementuak beraien artean ahalik eta antzekoenak izatea lortuko da, eta gainontzeko taldeen elementuekiko ahalik eta ezberdinenak. Hori dela eta, berdintasun edo dibergentzia neurri bat definitzea ezinbestekoa izango da elementu ezberdinak talde batean edo bestean sailkatzen joateko.

Egin behar den analisia, sailkapen algoritmo batez osatua egongo da, partizio baten edo gehiagoren lorpena ahalbidetuko duena. Egin behar den prozesu osoa, hurrengo eskemari

jarraituz eraiki daiteke:

- *n* elementuez osatutako multzo batetik abiatuko gara.
- Sailkapena egiteko algoritmo bat aukeratuko da, elementuen arteko multzoen egitura zehaztuko duena.
- Kidetasun irizpide bat ezarriko dugu, kidetasun edo distantzia matrize bat lortzeko, elementuen arteko antzekotasunak erlazionatzen lagunduko diguna.
- Egitura hori grafikoen bidez zehaztuko da, dendrogramak edo zuhaitz diagramen bitartez, adibidez, sailkapena hierarkikoa bada.

Lan honetan erabili diren algoritmoak, *aglomeratua* eta *K-means* izan dira. Beraz, horiek dira azalduko direnak.

3.4.1 Algoritmo hierarkikoa: Aglomeratua

Algoritmo hierarkikoek bi modutan egin dezakete lan. Modu bat, existitzen diren taldeak elkartzea da, talde berri bat osatzeko. Beste modua, existitzen den talde bat zatitzea da, bi talde berri sortzeko. Elkartze edo zatitze prozesu hau hurrenez hurren errepikatzen bada, datuen arteko distantzia minimizatuko edo antzekotasun neurriren bat maximizatuko da. Lan egiteko moduaren arabera, algoritmo hierarkikoak bi motatakoak dira: aglomeratuak edo zatitzaileak.

- Aglomeratuak: hasierako datu-multzoaren elementu bakoitzeko talde bat osatzen du. Hierarkia bat jarraituz, taldeek bat egiten dute hurrenez hurren. Taldeek bat egiten duten heinean, taldeen arteko homogeneitatea urritzen doa. Prozesuaren amaieran, sortu diren talde guztiak bakar batean biltzen dira.
- Zatitzaileak: aglomeratuen kontrako prozesua egiten du. Hasierako datu-multzoaren elementuak talde bakar batean biltzen ditu. Ondoren, hasierako talde handi hau zatitzen doa talde txikiagoak sortuz, eta, prozesuaren amaieran, datu-multzoaren elementu bakoitzeko talde bat egongo da.

Lan honetan aglomeratua erabili da. Hori dela eta, taldeak edo clusterrak batzeko erabili den estrategia edo kidetasun irizpidea azalduko da. Izan ere, taldekatzea zein modutan egingo den erabakitzeaz gain, taldeen edo clusterren arteko distantziak zein modutan

neurtuko diren zehaztu behar da. Erabili den estrategia edo distantziak neurtzeko metodoa, *batezbesteko haztatua (Weighted aritmethic average)* izan da.

Batezbesteko haztatua (WARD: Weighted aritmethic average)

Zenbait elementu cluster berean elkartzen direnean, informazio galera dago. Galera hori modu honetan neurtzen da: lehenengo, elementuak integratuko diren clusterraren batezbestekoa kalkulatzen da. Ondoren, elementu bakoitzaren eta batezbestekoaren arteko desbideratze karratua kalkulatzen da. Azkenik, desbideratze karratu guztien batura totala kalkulatzen da.

Sortzen diren talde berriek jatorrizko datuen ezaugarriak ez eraldatzeko, Ward zientzialariak hurrengoa proposatu zuen: Analisiaren pausu bakoitzean, talde-pareen bateratzea aintzat hartu, eta desbideratze karratuen batura total txikiena egiten duen talde-parea hautatu.

3.4.2 Algoritmo ez-hierarkikoa: K-means

Algoritmo ez-hierarkikoek analisia hasi baino lehen, cluster kupurua zehaztea exijitzen dute. Hori errazteko, multzo kopuru optimoa zein den adierazten duen indizeren bat edukitzen dute. Lan honetan erabili den metodoa, *K-means* izan da, elementuek zentroideekiko daukaten distantzia neurtzen duena.

K-means

n elementuez osatutako multzo bat *k* multzotan banatzen du. Talde bakoitza osatzen duten elementuen arteko batezbestekoa kalkulatzen da. Batezbesteko horri, zentroide deritzo. Zentroidea, talde bakoitzaren ordezkaria izango da.

Clustering metodo honek, hasieran *k* zentroide aleatorioki ezartzen ditu. Jarraian, elementu bakoitza gertuen duen zentroidera egokitzen du. Ondoren, zentroideen balioak egokitzen ditu zentroideari egokitu zaion elementuen batezbestekoa kalkulatuz. Azkenik, gertutasunaren arabera, elementuak zentroideen balio berrietara berriz egokitzen ditu.

3.5 Ikasketa gainbegiratua

Ikasketa automatikoaren algoritmo beste mota bat da. Lehenagotik etiketatutako adibideetatik abiatuz, funtzio bat sortzen du. Funtzio horrek, sistemako sarrera eta irteeraren arteko korrespondentzia bat sortzen du. Hau da, ondo identifikatutako behaketa multzo batez moldatzen den ikasketa da.

Iragarpenak egiteko, ezaguna den datu-multzo bat erabiltzen du. Ezaguna den datu-multzo horri, entrenamendu multzoa deritzo. Entrenamendu multzoa sarrerako-datuez eta espero diren emaitzez osatua dago. Iragarpenaren emaitza, zenbaki bat (problemas de regresion) edo klase bateko etiketa (problemas de clasificacion) bat izan daiteke. Maistasunez, datumultzoaren azpimultzo bat test moduan erabiltzen da iragarpenaren emaitza balioztatzeko. Entrenamendu multzoa zenbat eta handiagoa izan, orduan eta emaitza hobeak lortuko dira. Ikasketa gainbegiratua bi kategoriatan banatzen da:

- Sailkapena: Iragarpenaren emaitza etiketatua denean
- Erregresioa: Iragarpenaren emaitza zenbaki bat denean

Lan honetan erabili den kategoria edo algoritmoa, sailkapen algoritmoa izan da.

3.5.1 Sailkapen algoritmoa

Elementu bakoitza bi klase edo gehiagoren artean etiketatzen saiatzen da. Iragarpen ereduak sortzen dituzte klase bati dagozkion etiketak eta klase horren ezaugarriak dituzten datuetatik abiatuz. Iragarpen eredu hauek, datuen ezaugarrietatik ikasitakoa datu berrien gainean aplikatzen dute, datu berrien etiketak aurreikusteko. Bi klaseren arteko aukeraketari sailkapen bitarra deritzo. Bi klase baino gehiagoren arteko aukeraketari klaseaniztasun sailkapena deritzo.

Sailkapena egiteko, sailkatzaileak behar dira. Lan honetan erabili den sailkatzailea, *konfiantza elipsoidea* izan da.

3.5.2 Sailkatzailea

Etiketatuta dauden datu-multzo baten azpimultzo bat hautatzea, eta, hortik habiatuta, sarrerako beste edozein daturi etiketa bat esleitzeko gai den erregela bat definitzea, sailkatzaile bat inplementatzeko modu bat da. Honen adibide bat, mezu elektroniko bat "spam"edo "ez spam"sailkapenaren arabera sailkatzea izan daiteke. Edo, gaixo bati diagnosi bat esleitzea gaixoaren ezaugarriak kontuan harturik (Generoa, presio arteriala, sintoma batzuen gabezia ...). Orokorrean, sailkapena ereduen ikuskapenaren [2] adibide bat da.

Sailkatzaile gehienek, bi klasetako behaketak sailkatzeko balio dute. Baina, lan honetan, klase bakarreko behaketak kontuan hartu ziren. Hori dela era, klase bakarreko behaketak behar zituen sailkatzaile sinplea behar zen. Hori kontuan hartuz, *konfiantza elipsoidea* sailkatzailetzat erabili zen.

Konfiantza elipsoidea

Klase bakarreko behaketekin eraikiko den sailkatzailea da. Sailkatzaile honekin lortuko diren emaitzak balioztatzeko, balioztatze-gurutzaketa teknika erabiliko da. Etiketa bakarreko X multzoa emanik, sailkatzailearen formula hurrengoa da:

$$(x_o - \bar{X})' S^{-1} (x_o - \bar{X}) \leq \chi^2_{\frac{2,0.95}{5}}$$

non $\bar{X}$, X multzoaren batezbestekoa eta S^{-1}, X multzoaren bariantza-kobariantza matrizearen alderantzizkoa den. Ezkerraldean agertzen den adierazpena, Mahalanobis distantziaren formula da.

x_o alea, sailkatu beharko den ale berria izango da. "Emaitza"$\leq \chi^2_{\frac{2,0.95}{5}}$ baldin bada, x_o alea Xren motatakoa izango da. Bestela, beste mota batekoa.

3.5.3 Balioztatze gurutzaketa

Eredu estatistikoen emaitzak ebaluatzeko balio duen teknika da, entrenamenduzko eta test azpimultzoen independentzia bermatzen duena. Orokorrean, eredu baten test errorearen

[2] Ereduen ikuskapena zientzia bat da. Ingeniaritzako, konputazioko eta matematikako objektu fisiko edo abstraktuen prozesuez arduratzen da. Informazioa aterata, objektu horien multzoen artean propietateak ezartzea da helburua.

neurri bat estimatzeko balio du, eta horri esker, ereduaren iragarritasuna ebaluatu daiteke.

Erabiliko den metodologiaren ordena azalduko da:

1. Hasieran aipatu bezala, datu-multzoa bi ataletan banatu behar da:
   - Entrenamendua (E): Eredu bat entrenatzeko azpimultzoa
   - Test (T): Entrenatutako eredua probatzeko azpimultzoa
2. Ondoren, sailkapenerako erabiliko den teknika entrenamenduzko multzoari aplikatuko zaio, eta sailkatzaile bat sortuko da. Lan honetan erabili zen sailkatzailea, *Konfidantza Elipsoidea* izan zen, 3.5.2 atalean azaldu dena.
3. Azkenik, sailkatzaileak test multzoan egiten duen errorea kalkulatuko da.

Balioztatze gurutzaketaren metodo ezberdinen ezberdintasunak, entrenamendu eta test multzoak sortzerako moduan dago. Lan honetan, *leave-one-out* metodoa erabili denez, hori da azalduko dena.

Leave-one-out

Leave-one-out (LOO) metodo iteratibo bat da. Metodo hau hasieratzeko, entrenamenduzko multzoan behaketa guztiak kontuan hartzen dira bat izan ezik. Kontuan hartu ez dena, test multzorako erabiltzen da. Test errorearen indizea test multzoan erabili den behaketaren araberakoa da. Hori dela eta, prozesu hau behaketa guztiekin errepikatu behar da, iterazio bakoitzean test multzoan behaketa ezberdina erabiliz. Hau da, $i = 1, \ldots, n$ behaketa badaude, iterazio bakoitzean i ezberdin bat testeatuko da (eta gainontzekoak entrenamenduzko multzoan erabiliko dira), eta i bakoitzerako test errorea kalkulatuko da, $i = n$ kasuaren test errorea kalkulatu arte. Hortaz, LOO metodoaren bitartez kalkulatzen den test errorearen indizea i erroreen batezbestekoa estimazioa da. Errorearen kalkuluaren formula honako hau da:

$$CV_n = \frac{1}{n}\sum_{i=1}^{n}(MSE_i) \tag{3.14}$$

non MSE errorearen batezbesteko koadratikoa den.

Demagun $\hat{y}_{(i)}$ sailkatzaileak emandako erantzuna dela eta $y_{(i)}$ sailkatu nahi den datua. Hori kontuan hartuz, LOO metodoaren erantzuna $\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_{(i)} - y_{(i)})^2$ izango da, non:

$$y_{(i)} = \begin{cases} 1, \text{ 1 klasekoa bada} \\ 0, \text{ bestela} \end{cases}$$

Laburbilduz, *Leave-one-out* teknika erabiliz, hau da jarraituko den prozedura, entrenamendurako n datu baldin badaude, hurrengo pausuak n aldiz errepikatuko direla kontuan izanik:

- datu-multzoko azpimultzo bat test multzorako erreserbatuko da.
- Geratzen diren $n-1$ datuak entrenamendurako utziko dira.
- n. datuarekin testa egin, sailkatzailea erabiliz (emaitza asmatzea edo huts egitea bakarrik izango da).

LOO metodoari esker, behaketak entrenamendu eta test multzoetan zoriz banatzen direnean sortzen den aldakortasuna murrizten du. Hori horrela da LOO metodoaren prozesuaren amaieran datu guztiak testekoak eta entrenamendukoak izan direlako. Baina, metodo honek desabantaila bat dauka. Desabantaila hori daukan konputazio kostua da. Izan ere, prozesua behaketa kopuru adina aldiz errepikatu behar da.

4. KAPITULUA

Erabilitako tresnak

Atal honetan proiektua garatzeko erabili ziren tresnak azalduko dira. Kasu honetan, R lengoaia, R studio softwarea eta bertan inplementatuta zuden funtzioak erabili ziren. Beraz R eta R studio zer diren eta lan hau aurrera eramateko R studiotik erabili ziren funtzioak/-tresnak azalduko dira.

4.1 R lengoaia

R analisi estatistikoaren ingurunera bideratua dagoen programazio lengoaia da. S lengoaiaren software librearen birinplementazioa egiterakoan R lengoaia sortu zen. Beste lengoaia batzuk liburutegiak erabiltzen dituzte, baina R, aldiz, doaneko paketeak deskargatuz eta erabiliz funtzionatzen du. GNU lizentzia dauka, hau da, librea, doakoa eta kode irekiko lengoaia da, beraz edonork erabili dezake. Eskaintzen dituen irudikapen eta estatistika tresnei esker, ikerkuntza zientifikoan gehien erabiltzen den programazio lengoaia da. Gainera, kalkuluak egiteko ahalmen handia dauka. Hori dela eta, oso ezaguna da datu meatzaritza, bio-medikuntza ikerketa, ikasketa automatikoan eta bio-informatika motako arloetan.

Orokorrean, hauek dira R lengoaiak dituen ezaugarriak:

- Erreminta estatistiko ugari eskaintzen ditu:
  - Eredu linealak eta ez-linealak

  - Test estatistikoak
  - Algoritmo sailkatzaileak eta elkartze algoritmoak

- Irudikapen ahalmen handia dauka, eta, horri esker, kalitate handiko grafikoak sor daitezke. Pakete ugari existitzen dira grafiko mota ezberdin asko egiteko aukera zabala eskaintzen dituztenak.

- Integragarria da. Hau da, datu-base ezberdinetan biltegiratuak dauden datuak atzitzeko aukera eskaintzen du. Gainera, hainbat pakete existitzen dira R beste lengoaia batzuekin (Perl, Ruby eta Python, adibidez) elkarri eragitea eta beraiekin objektuak trukatzea ahalbidetzen dutenak.

- Objektuei orientatutako [1] lengoaia izateaz gain, erabiltzaileek lengoaia hedatu dezakete euren funtzio propioak erabiliz.

- Gure programan definitzen diren objektu guztiak gure makinaren memorian gordetzen ditu. Hori dela eta, garrantzitsua da memoria nola kudeatzen duen ulertzea, gure kodea ahalik eta gehien optimizatzeko. Horri esker, gure makina motelduko duten beharrezkoak ez diren objektuen kopiak ekidin daitezke.

- R lengoaia interpretatua [2] da. Hau da, kodea ez da konpilatu behar, R-ren interpretatzaileak zuzenean exekutatzen du. Interpretatzaile hori, programa bat izan ohi da. Bere ardura, "giza"semantikan idatzita dagoen agindu bakoitza makina kodera (ordenagailuaren CPU-aren aginduak) itzultzea da. Lan honetan erabili den interpretatzailea R studio izan da.

4.1.1 R studio

R programazio lengoaiarentzako integratutako garapen ingurunea (*IDE*) da, konputazio estatistikoa eta grafikoak erabiltzea ahalbidetzen duena. Windows, Mac eta Linux sistema eragileentzako eskuragarri dago, bai eta RStudio Server edo RStudio Server Pro zerbitzarietara konektatuta dauden nabigatzaileentzako ere. Kontsola bat, kodearen exekuzioari laguntza ematen dion sintaxi editore bat eta lan memoriaren arazketarako eta kudeaketarako hainbat erreminta dauzka.

[1] Datu mota ezberdinen definizioak, datu mota horien gaineko operazioak eta datu mota horien instantziak ahalbidetzen dituen edozein lengoaiari deritzo

[2] Konpiladore batez prozesatuak izan behar ez duten lengoaiei esaten zaie. Konputagailua programatzaileak idazten dituen agindu segida exekutatzeko gai da hauek irakurri eta itzuli beharrik izan gabe.

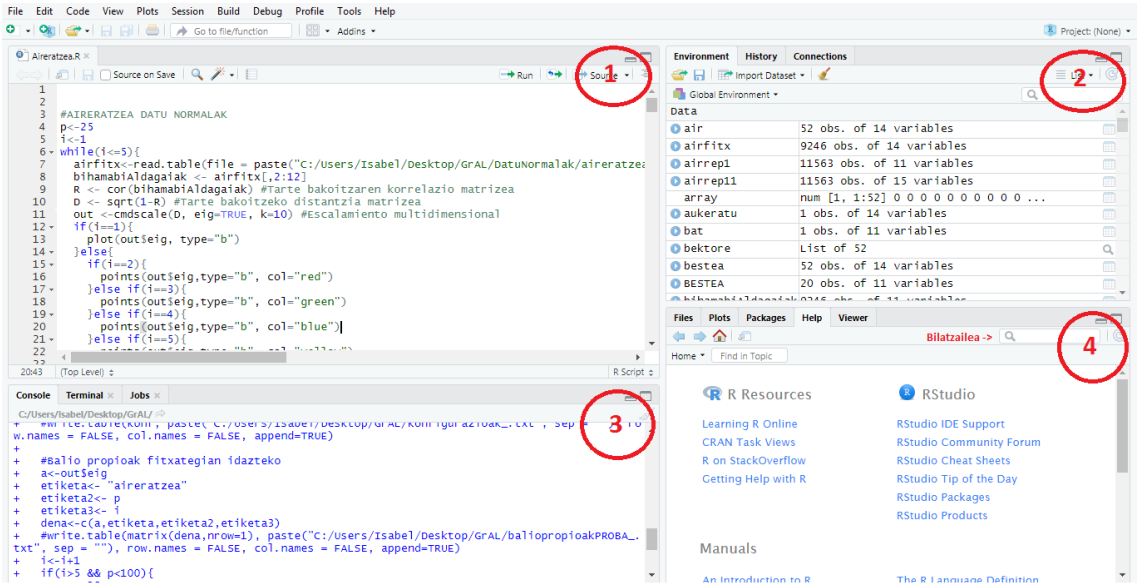
Orokorrean, hauek dira R studiok dituen ezaugarri nagusienak:

- Bakarrik R lengoaiarako sortu den integratutako garapen ingurunea (IDE)
  - Nabarmendutako sintaxia, kode osatua eta galera adimentsua.
  - Iturburu-kodearen editoretik R kodea zuzenean exekutatu.
  - Definituta dauden funtzioekiko jauzi azkarra.
- Lankidetza
  - Integratutako euskarri eta dokumentazioa.
  - Proiektuen bitartez askotariko direktorioen administrazio erraza.
  - Datuen bisorea eta lan eremuen nabigazioa.
- Autoretza eta arazketa ahaltsuak
  - Araztaile elkarreragilea erroreak aurkitzeko eta erroreak azkar zuzentzeko.
  - Garapen erreminta zabala.
  - Sweave [3] eta R Markdown [4] autoretza.

Hurrengo irudiaren bitartez, R studioko interfazea eta bertan atzigarri dauden aukerak azalduko dira:

[3] R lengoaiaren osagaia da, kodearen integrazioa ahalbidetzen duena LaTeX-ekin edo LyX-ekin idatzitako dokumentuetan. Hau da, LaTeX editorearekin sortutako dokumentu arrunt batean testua eta kodea nahasi daitezke Sweave funtzioari esker.

[4] Dokumentuen sorkuntza, aurkezpen dinamikoak eta R-ren informeak sortzea ahalbidetzen duen formatu bat da. Informe edo dokumentu horiek, testua zein R kodea eta emaitzak biltzen dituzte. HTML, PDF eta Word dokumentuak sortzeko sintaxi arrunt bateko formatu bat da.



4.1 Irudia: R studioko interfazea

Lau leiho ezberdin daude, eta goiko aldean R studion dauden aukera ezberdinak hautatzeko barra dago.

1. Sintaxi editorea
   Leiho honetan sintaxia editatzen da ondoren exekutatu ahal izateko. Hau da, bertan kodea idazten da. Ez da ezer gertatuko **Run** botoia sakatu arte.

2. Programaren lan eremua
   Eremu honetan, gure makinaren memorian gordetzen dituen datu-multzoak eta objektuak (emaitzak, aldagaiak, grafikoak etab) agertzen dira kodea exekutatu eta gero.

3. Kontsola
   R softwarearen oinarrizko bertsioari dagokio. Softwareak sintaxi editorean idatzitako eragiketak exekutatzen ditu.

4. Hainbat azpi-leiho dauzka
   Files fitxak, programarekin lan egindako fitxategien historiala ikustea ahalbidetzen du. **Plots** fitxak, programan zehar sortutako grafikoak ikustea ahalbidetzen du. Eta nahi izanez gero, sortzen diren grafiko horiek deskargatu daitezke. **Packages** fitxak, deskargatutako paketeak eta disko gogorrean gordetako paketeak ikustea ahalbidetzen du, bai eta pakete hauen eguneratzeen edo instalazioen kudeaketa ere. **Help**

fitxak, CRAN (Comprehensive R Archive Network) atzitzea ahalbidetzen du, Internetera konexioa baldin baduzu. CRAN, R studioko softwarearen orri ofiziala da, eta programarako hainbat baliabide ezberdin eskaintzen ditu: Erabiltzailearentzako eskuliburuak, on-line kurtsoak, informazio orokorra, paketeen deskargak, deskargatutako paketeen informazioa, funtzioen erabilpenak etab. Fitxa hau oso erabilgarria da. Fitxa horretan dagoen bilatzailea erabiliz, gure konputagailuan instalatuak dauden pakete (eta bere funtzioak) ezberdinen eskuliburuak oso azkar atzitu daitezke (Interneteko konexioaren beharrik izan gabe), nola erabiltzen diren jakiteko. **Viewer** fitxak, *markdown* motako funtzionalitateen bidez sortutako txostenen emaitzak erakusten ditu.

4.2 R studioko funtzioak

Azpiatal honetan, proiektua aurrera eramateko R studioko softwareko pakete ezberdinetan inplementatuta dauden funtzioetatik, erabili ziren funtzio aipagarrienak azalduko dira.

4.2.1 cor

Rko **cor** funtzioa, korrelazio-matrize bat sortzeko erabili daiteke. Funtzioaren erabilpen arrunta hurrengoa da:

$$cor(x, method = c(\text{"}person\text{"}, \text{"}kendall\text{"}, \text{"}superman\text{"}))$$

- **x**: zenbakiez osatutako matrizea edo datu-egitura bat
- **method**: kalkulatu behar diren korrelazio-koefizienteak zein metodorekin kalkulatuko diren adierazten du. Metodo lehenetsia *Pearsonen* korrelazio koefizientea 3.1.3 da.

Kendall eta *Spearman* metodoak, korrelazio analisi ez-parametrikoak egiteko erabiltzen dira. Korrelazio analisi ez-parametrikoetan, ez da beharrezkoa datu-multzoa aldagaietan bereizita egotea. Lan honetako datu-multzoa aldiz, aldagai batzuez bereizita zegoen. Hori dela eta, *Pearsonen* metodoa erabili zen korrelazio koefizienteak kalkulatzeko.

cor() funtzioari esker, ondoren kalkulatuko zen dimentsio anitzeko mailaketan beharrezkoa zen korrelazio-matrizearen koefizienteak kalkulatu ahal izan ziren.

4.2.2 cmdscale

Datu-multzo baten gainean dimentsio anitzeko mailaketa kalkulatzeko erabiltzen da. Funtzioaren erabilpen arrunta hurrengoa da:

$$cmdscale(d, k = 2, eig = FALSE, add = FALSE, x.ret = FALSE, list. = eig \parallel add \parallel x.ret)$$

- **d**: distantzia edo korrelazio matrizea.
- **k**: espazioaren gehienezko dimentsio kopurua datuak adierazteko
- **eig**: 3.2.2 atalean azaltzen den Q matrizearen balio propioak itzuli nahi diren edo ez adierazten du. $eig = TRUE$ idazten bada, emaitzan balio propioak agertuko dira.
- **add**: konstante gehigarri bat kalkulatu behar den eta 3.2.2 atalean azaltzen den Q matrizearen diagonalean gehitu behar den adierazten du.
- **x.ret** 3.2.2 atalean agertzen den PDP matrizea da. $x.ret = TRUE$ idazten bada, emaitzan itzuliko du matrize horren balioa.
- **list.**: lista bat edo bakarrik 3.2.2 atalean azaldutako koordenatu nagusiez osatutako Y matrizea itzuli behar den adierazten du.

list. aldagaiaren balio lehenetsia FALSE denez, besterik ezean koordenatu nagusiez osatutako matrizea bakarrik itzuliko du irteeran. $list. = TRUE$ jarrita aldiz, **list.** aldagaia osatzen duten osagai guztiak irteeran itzuliko dira. Hauek dira **list.** dituen osagaiak:

- **points**: koordenatu nagusiez osatutako matrizea 3.2.2 atalean agertzen den Y matrizea dena.
- **eig**: lehen adierazitako Q matrizearen balio propioak.
- **x**: lehen adierazitako PDP matrizea.
- **ac**: konstante gehigarria, 0 baldin $add = FALSE$ bada.

Beraz, funtzio honi esker dimentsio anitzeko mailaketarekin lortzen den distantzia-matrizetik abiatuz, koordenatu nagusiez osatutako matrizea lortuko da.

4.2.3 vegan::procrustes

vegan paketeko funtzio honek, konfigurazio baten biraketa egiten du beste konfigurazio batekiko ahalik eta gehien gerturatzeko. Funtzio honi esker, 3.3 atalean azaltzen den Procrustes estatistikoaren koefizientea lortzen da. Funtzio honen erabilpena hurrengoa da:

$$procrustes(X, Y)$$

- **X**: helburu matrizea
- **Y**: biratu behar den matrizea

Funtzio horrek aldagai gehiago hartzen ditu sarreran, baina bi aldagai horiek jartzearekin nahikoa izan da, praktikan erabili ziren bi matrizeek dimentsio eta metrika berdina zeukatelako.

Irteeran, hainbat datu ezberdin bueltatzen ditu, esaterako: **Yrot** (Y matrize biratua), **X** (helburu matrizea), **translation** (jatorriaren translazioa), **scale** (eskala faktorea) etab. Baina funtzio honek irteeran bueltatzen dituen balio guztien artetik, kontuan hartu zena hurrengoa da:

- **ss**: Y matrizearen koordenatuen transformazioen desbiderazio karratuen batura minimoa X matrizearen koordenatuekiko. Hau da, Procrustes estatistikoaren balioa itzultzen du.

Beraz, argi geratu ez bada, funtzio honekin lan honetan garatu zen Procrustesen analisia 3.3 egin zen.

4.2.4 hclust

Datu-multzo bateko Cluster analisi hierarkikoa **??** eta multzo hori aztertzeko metodoak aplikatu ahal izateko balio duen funtzioa. Funtzio honen erabilpena hurrengoa da:

$$hclust(d, method = "ward.D2")$$

- **d**: distanzia-matrizea

- **method**: cluster analisi hierarkikoa aplikatzeko erabiltzen den metodoa. Hainbat metodo daude: *ward.D*, *ward.D2*, *average*, *median* etab. Lan honetan erabili den metodoa *ward.D2* izan da. Metodo honetan, helburu funtzioa karratuen baturaren edo bariantzaren errorea da.

Funtzio honen irteera, **hclust** klasearen objektu bat da, clustering prozesuan sortzen den zuhaitza deskribatzen duena. Objektu hori lista bat da. Listak dituen osagaietatik, hauek dira aipagarrienak:

- **merge**: $n-1 \times 2$ dimentsioko matrize bat da. *Merge*-ren i.lerroak, clusteringaren i.pausoan ematen den taldeen (clusterren) bateratzea deskribatzen du. Lerroaren j.elementua negatiboa bada, orduan $-j$.elementua pausu honetan bat-egin du. j.elementua positiboa bada, orduan bat-egitea algoritmoaren aurreko j.pausuan sortutako clusterrarekin izan da.
- **height**: taldekatzearen altuera. Hau da, taldekatze metodoari dagokion irizpide balioa metaketa egiteko.
- **order**: jatorrizko behaketen permutazioekin osatutako bektorea.
- **label**: taldekatutako elementu bakoitzerako etiketa.

Aldagai horiei esker, dendrograma bat irudikatu ahalko da, behaketen sailkapena adierazten duena.

4.2.5 pam

Datu-multzoa k clusteretan (taldeetan) banatzen ditu. *K-mean* metodoaren bertsio sendoagoa da. Funtzioak hainbat aldagai ditu sarreran, baina hurrengo biak erabili dira, beraz erabilpena hurrengoa da:

$$pam(x,k)$$

- **x**: distantzia-matrizea.
- **k**: Zenbaki positibo bat, cluster (talde) kopurua adierazten duena. Zenbakia behaketa kopurua baino txikiagoa izan behar du.

Funtzioaren irteera *pam* klaseko objektu bat da, *clustering*ak adierazten dituena. Objektu hori lista bat da, hainbat osagai dituena. Baina lan honetan lista horretatik kontuan hartu den osagai bakarra hau izan zen:

- **clustering**: Zenbaki positiboez osatutako *n* luzerako bektore bat da. *n* behaketa kopurua da. Behaketa bakoitzari zein cluster zenbaki dagokion adierazten du, hau da, behaketa bakoitzari dagokion taldearen identifikazioa.

4.2.6 silhouette

Funtzio honek, datu-multzorako cluster (talde, k) egokiena zein den identifikatzen laguntzen du. Siluetaren batezbestekoaren balioa erabiltzen da cluster zenbaki optimoena jakiteko. Funtzioaren erabilpena hurrengoa da:

$$silhouette(x, dist)$$

- **x**: k cluster zenbaki positiboez osatutako bektorea.
- **dist**: distantzia-matrize bat.

Funtzio honen irteera, *silhouette* klaseko objektu bat, *sil*, itzultzen du. Objektu hori $n \times 3$ dimentsioko matrize bat da. i.behaketa bakoitzerako, $sil[i,]$, i zein klasekoa den, bere ondokoak (bizilagunak) zeintzuk diren eta bere siluetaren zabalera zein den adierazten du. Behaketa guztien siluetaren zabaleraren balioa kontuan hartzea oso garrantzitsua izango da, balio horien batezbestekoa egiten delako cluster optimoaren balioa zein den ezagutzeko.

Aipatu diren azken hiru funtzioak lan honetako datu-multzoaren gainean clustering prozesua aplikatzeko eta datuak sailkatu ahal izateko baliogarriak izan ziren. Elementuen arteko loturak ikustea ahalbidetu zuten.

4.3 Eclipse eta Java

Lan honen hasierako urratsetan, Eclipse ingurunea erabili zen Java lengoaian programatu ahal izateko. Izan ere, hasiera batean ez ziren R lengoaia eta R studio ondo menderatzen, eta Java lengoaia hobeto menderatzen zen.

4.4 Overleaf

Dokumentuak idazteko Interneteko konexioa behar duen onlineko Latex editorea da. Dokumentu zientifikoak idazteko tresna oso baliogarria da. Online denez, edozein makinatik erabili daiteke, Latex editore ordenagailuan instalatuta eduki beharrik izan gabe. Horregatik online editore hau erabili zen memoria idazteko.

5. KAPITULUA

Garapena

Proiektu hau aurrera eramateko eman ziren pausoak urratsez urrats azalduko dira eranskin honetan.

5.1 Hasierako datuak

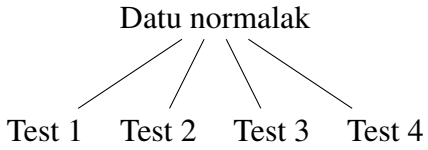
Hasteko, dronaren motorraren datuak jasota zeuzkan karpeta *www.ehu.eus/zaindegi* plataformatik deskargatu zen. Karpeta horretan zeuden datuak, bi karpeta ezberdinetan banatuak zeuden. Karpeta batean, datu normalak zeuden, hau da, eraldaketarik jaso ez zituzten datuak. Bestean, datu anomaloak zeuden, hau da, eraldaketak jaso zituzten datuak.

Horrez gain, datu anomaloak, beste bi karpetatan banatzen ziren. Lehen karpetak, *Zarata* zuen izena. Bertan zeuden datuen *azelerazio* aldagaiari %5eko eta %10eko zarata (banaketa uniformea) gehitu zitzaizkion. Bigarren karpetak, *Desoreka* zuen izena. Karpeta horretan zeuden datuak, dronaren helizeari desoreka bat egindako prozesuan bildutako datuak ziren.

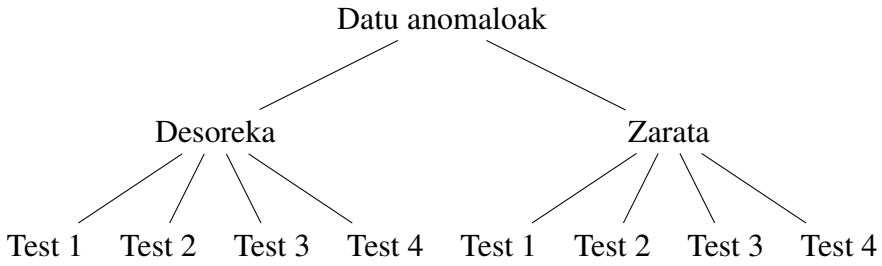
Gainera, dronaren motorraren datu normalak zein datu anomaloak, dronaren motorraren potentzia ezberdinei zegozkien. Izan ere, dronarekin frogak egin zirenean, dronaren motorraren potentzia ezberdinekin egin ziren. Dronaren potentzia **%25**, **%50**, **%75** eta **%100** ziren. Potentzia bakoitza, Test kasu bati zegokion:

- %25-eko potentzia - Test 1
- %50-eko potentzia - Test 2
- %75-eko potentzia - Test 3
- %100-eko potentzia - Test 4

Beraz, hasieran datuak horrela banatuak zeuden:



5.1 Irudia: Hasierako datu normalak
(Zuhaitzaren nodo bakoitza karpeta bat da)



5.2 Irudia: Hasierako datu anomaloak
(Zuhaitzaren nodo bakoitza karpeta bat da)

Dronaren hiru egoera ezberdinekin egin ziren frogak. Hiru egoera horiek, **aireratzea**, **lurreratzea**, eta **hoztea** ziren. 5.1 eta 5.2 irudietan agertzen diren Test karpeta bakoitzaren barruan zegoen fitxategiak hiru egoerei buruzko datuak biltzen zituen. Gainera, egoera eta Test (potentzia) ezberdin bakoitzeko, froga bat baino gehiago egin ziren, beraz errepikapen bat baino gehiago egin zen egoera eta test (potentzia) bakoitzeko. Errepikapenak ere, Test karpeta bakoitzean zegoen fitxategian bilduta zeuden.

Horietako fitxategi bakoitzean zeuden datuak, 12 zutabeetan banatzen ziren (dronari buruzko informazioa biltzen zituzten aldagaiak ziren):

1. Denbora [s]
2. Azelerazioa [0-100%]

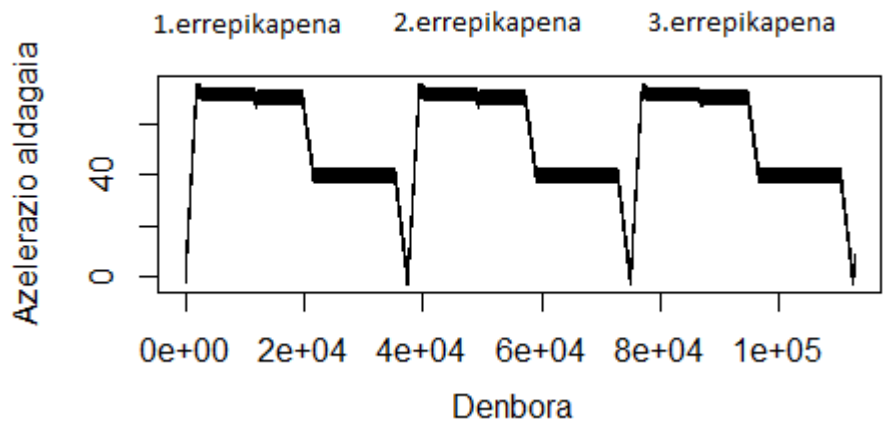
3. Motorraren tenperatura [ºC]
4. Bultzada [N]
5. DC bus korronte [A]
6. U korronte fasea [A]
7. V korronte fasea [A]
8. W korronte fasea [A]
9. Bataz-besteko abiadura [RPM]
10. Bat-bateko abiadura [RPM]
11. PWM 250 Hz lan-zikloa [ez dimentsionala]
12. ESC tenperatura [ºC]

5.2 Datuak egoeraka banatu

Informazio hori guztia edukita, datuak egoeraka (aireratzea, lurreratzea eta hoztea) banatu ziren. Lortutako emaitzekin, datu normalen eta ez-normalen arteko erlazioak ezagutu behar ziren eta egoera anomaloak baldin bazeuden, detektatu.

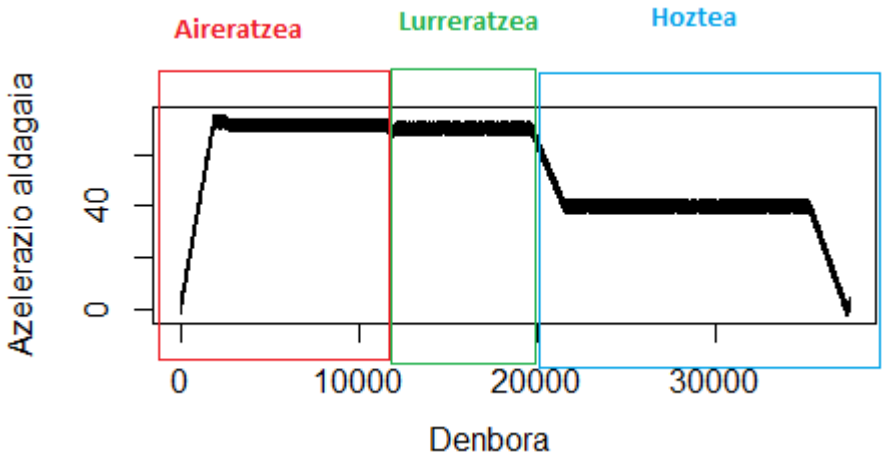
Datuak egoeraka banatzeko, R studio eta Eclipse erabili ziren. R studio, fitxategiko datuen antolamendua ikusteko erabili zen. Horretarako, datu horiek R studio erabiliz grafiko batean irudikatu ziren. Horrela, fitxategi horretan zenbat errepikapen zeuden eta egoera bakoitzaren datuak fitxategiko zein lerroetatik zein lerroetara ziren identifikatu ahal izan zen. Behin hori identifikatuta, Java erabili zen datuak egoeraka banatzeko.

Aurrekoa paragrafoan azaldutakoa, adibide bat jarrita hobeto ulertuko da. Adibidez, baldintza hau daukaten datuen fitxategia hartuko dugu: Datu anomaloak, dronaren motorreko potentzia %25 eta azelerazio aldagaian %5eko zarata gehitu zitzaiena. Fitxategi hori, 5.2 irudian ikusten den **Datu anomaloak -> Zarata -> Test 1** karpetan zegoen. R studio erabiliz, fitxategi horretako datuak irakurri eta horiek irudikatu ziren, eta hurrengo grafikoa lortu zen:



5.3 Irudia: Datu anomaloen Zarata karpetako Test 1 karpetan dagoen fitxategiaren azelerazio aldagaiaren datuen irudikapena. Fitxategi horretako datuen azelerazio aldagaiari %5eko zarata gehitu zitzaion.

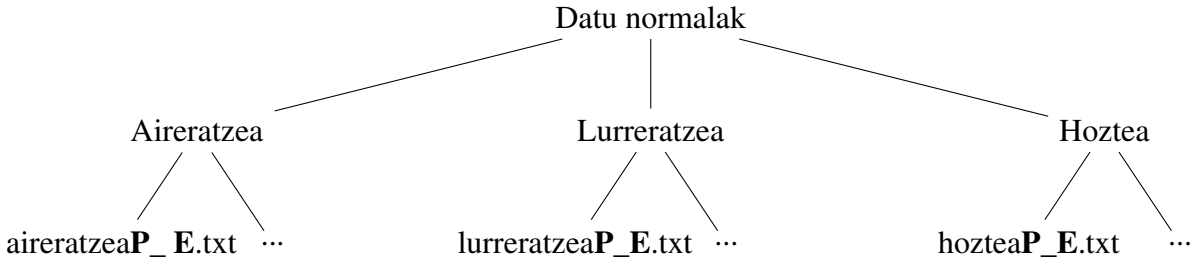
5.3 irudia erreparatuz, dronaren motorraren potentzia %25ekoa (Test 1 zelako) zenean eta azelerazio aldagaiari %5eko zarata gehitu zitzaionean, frogak hiru aldiz egin zirela ikus zitekeen. Beraz, Java erabiliz, fitxategi hori hirutan banatu zen, errepikapen bakoitzeko fitxategi berri bat. Ondoren, hiru fitxategi hauen datuak berriz irudikatu behar izan ziren egoera bakoitzari zegozkien datuak identifikatzeko. Hurrengo irudian, lehen errepikapenari zegokion datuen irudikapena ikusiko da:



5.4 Irudia: Datu anomaloen Zarata karpetako Test 1 karpetan dagoen fitxategia hirutan zatitu ondoren (hiru errepikapen zituelako), lehenengo errepikapenari dagokion datuen irudikapena.

5.4 irudian ikusten den bezala, egoera bakoitzari zegokion datuen nondik-norakoa jakin ondoren, Java erabili zen datuak egoeraka banatzeko.

5.1 eta 5.2 irudietan agertzen den Test karpeta bakoitzaren fitxategiarekin gauza bera egin zen, datu guztiak egoeraka banatu arte. Izan ere, dronaren egoera (aireratzea, lurreratzea eta hoztea) bakoitzarekiko datu normalen eta ez-normalen arteko erlazioak ezagutzea eta egoera anomaloak baldin bazeuden, detektatzea lan honen helburua izan zelako. Beraz, datuak egoeraka banatzea lortu zenean, amaieran datuak horrela banatuak zeuden:



5.5 Irudia: Amaierako datu **normalen** antolaketa

Zuhaitzaren amaieran dauden nodoei, umerik ez daukatenei alegia, hostoak deritze. Hostoetan agertzen direnak fitxategiak dira. Zuhaitzaren gainontzeko nodoak, karpetak dira. Fitxategien izena modu honetan antolatu zen:

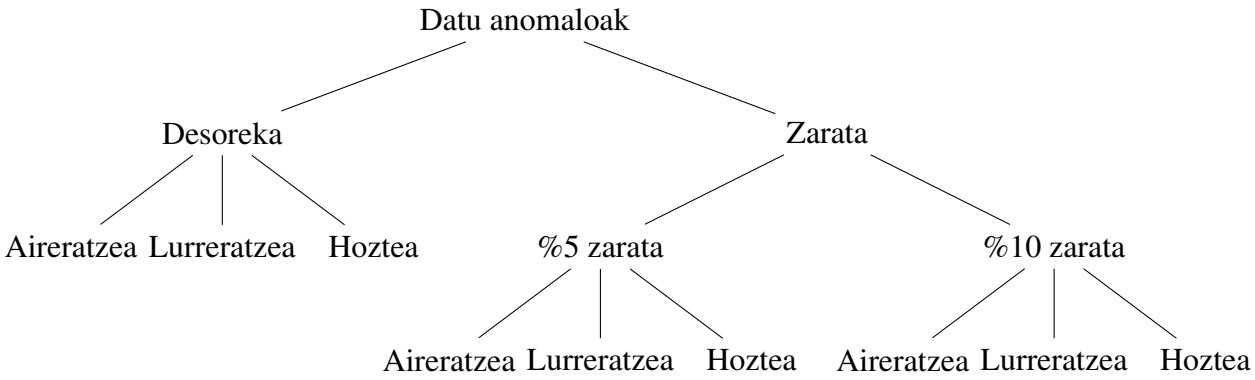
*egoera**P_E**.txt*

- Egoera aireratzea zenean: aireratzea*P_E*.txt
- Egoera lurreratzea zenean: lurreratzea*P_E*.txt
- Egoera hoztea zenean: hoztea*P_E*.txt
- **P**: motorraren potentzia adierazten du. **P**ren balioak hauek ziren: 25, 50, 75 edo 100.
- **E**: zenbatgarren errepikapena den adierazten du. **E**ren balioak hauek ziren: 1,2,3,4 edo 5 .

Datu normalei zegokionez, egoera (aireratzea, lurreratzea eta hoztea) bakoitzeko **20** kasu ezberdin zeuden. 5.1 taulan egoera bakoitzeko guztira zenbat kasu zeuden hobeto adieraziko da:

Egoera	Motorraren potentzia	Froga kopurua = Errepikapenak
Aireratzea	%25	5
	%50	5
	%75	5
	%100	5
Guztira		**20**
Lurreratzea	%25	5
	%50	5
	%75	5
	%100	5
Guztira		**20**
Hoztea	%25	5
	%50	5
	%75	5
	%100	5
Guztira		**20**

5.1 Taula: Egoera bakoitzeko kasu kopurua datu **normalei** dagokionez



5.6 Irudia: Amaierako datu **anomaloen** antolaketa

Zuhaitzaren nodo guztiak karpetak dira. Zuhaitzaren hostoetan (umerik ez daukaten nodoak, azken lerroan agertzen diren nodoak) agertzen den karpeta bakoitzaren barruan, fitxategiak zeuden. Fitxategi horiek modu honetan antolatu ziren:

*egoera**P_E**.txt*

- Egoera aireratzea + desoreka zenean: desaireratzea*P_E*.txt
- Egoera aireratzea + %5 zarata zenean: zarata5aireratzea*P_E*.txt
- Egoera aireratzea + %10 zarata zenean: zarata10aireratzea*P_E*.txt

- Egoera lurreratzea + desoreka zenean: deslurreratzea*P_E*.txt
- Egoera lurreratzea + %5 zarata zenean: zarata5lurreratzea*P_E*.txt
- Egoera lurreratzea + %10 zarata zenean: zarata10lurreratzea*P_E*.txt
- Egoera hoztea + desoreka zenean: deshoztea*P_E*.txt
- Egoera hoztea + %5 zarata zenean: zarata5hoztea*P_E*.txt
- Egoera hoztea + %10 zarata zenean: zarata10hoztea*P_E*.txt
- **P**: motorraren potentzia adierazten du. **P**ren balioak hauek ziren: 25, 50, 75 edo 100.
- **E**: zenbatgarren errepikapena den adierazten du. **E**ren balioak hauek ziren: 1,2 edo 3.

Datu anomaloei zegokionez, egoera (aireratzea, lurreratzea eta hoztea) bakoitzeko **32** kasu ezberdin zeuden. 5.2 taulan egoera bakoitzeko guztira zenbat kasu zeuden hobeto adieraziko da:

Egoera	Datuekiko eraldaketa	Motorraren potentzia	Errepikapenak
Aireratzea	Helizean **desoreka** gehitu	%25	2
		%50	2
		%75	2
		%100	2
	%5 **zarata** gehitu	%25	3
		%50	3
		%75	3
		%100	3
	%10 **zarata** gehitu	%25	3
		%50	3
		%75	3
		%100	3
Guztira			**32**
Lurreratzea	Helizean **desoreka** gehitu	%25	2
		%50	2
		%75	2
		%100	2
	%5 **zarata** gehitu	%25	3
		%50	3
		%75	3
		%100	3
	%10 **zarata** gehitu	%25	3
		%50	3
		%75	3
		%100	3
Guztira			**32**
Hoztea	Helizean **desoreka** gehitu	%25	2
		%50	2
		%75	2
		%100	2
	%5 **zarata** gehitu	%25	3
		%50	3
		%75	3
		%100	3
	%10 **zarata** gehitu	%25	3
		%50	3
		%75	3
		%100	3
Guztira			**32**

5.2 Taula: Egoera bakoitzeko kasu kopurua datu **anomaloei** dagokionez

5.3 Datuen analisia

Guzti hau egin ondoren eta datuak egoeraka antolatu ondoren, R studio erabiliz datuen analisia egin zen. Egoera bakoitzerako, R script bat sortu zen: **aireratzea.R**, **lurreratzea.R** eta **hoztea.R**. Hiru scriptetan jarraitu zen prozedura eta metodologia berdina izan zenez, script bakarrean egin zena azaltzea nahikoa izango da. Adibidez, **aireratze** egoerari dagokion scripta aukeratuko da, **aireratzea.R**, eta bertan egin zena urratsez urrats azalduko da:

Azalpena hasi baino lehen, jarraitu zen prozedura osoa gainetik azalduko da, egin zenaren ikuspen orokor bat edukitzeko:

1. Datu-multzoa finkatu:
   Datu-multzoa $n \times p$ neurrikoa zen, non:

   - **n**: kasu bakoitzari zegokion fitxategiaren lerro kopuru totala zen. i lerro bakoitzean, p aldagaiei zegozkien datuak bilduta zeuden.
   - **p**: fitxategi bakoitzean zeuden zutabeak ziren, guztira 12 zeuden. Zutabe horiek, dronari buruzko informazioa biltzen zuten aldagaiak ziren (5.1 atalean aldagaiei buruzko xehetasunak daude). Aldagai horietatik, lehen aldagaia, denbora, ez zen kontuan hartu[1]. Hortaz, $p = 11$ zen.

2. Erabiliko zen distantzia erabaki:
   Egin zen lehen urratsa, dimentsio anitzeko mailaketa izan zen. Dimentsio anitzeko mailaketa, elementuen arteko distantzia edo antzekotasunak adierazten zituen matrize karratu batetik hasten zen, D. Hortaz, distantzia-matrize bat sortu behar zen, eta, distantzia-matrizea sortzeko, erabiliko zen distantzia erabaki behar zen. Hori dela eta, korrelazio-distantzia erabiltzea erabaki zen.

3. Dimentsio anitzeko mailaketa metodoa aplikatu:
   Guztira 52 kasu zeuden (20 kasu datu normalei dagokionez eta 32 kasu datu anomaloei dagokionez. 5.1 eta 5.2 tauletan ikus daiteke). Kasu bakoitzarekiko, dimentsio anitzeko mailaketa egin zen.
   Metodo hau aplikatuz, bi motatako emaitzak lortu ziren:

[1] OHARRA: "Denbora"aldagaiak, drona piztuta zeraman denbora segundoka biltzen zuen, baina ez zuen dronari buruzko informaziorik ematen. Hau da, "denbora"aldagaiak ez zuen drona osatzen zuten osagaiei buruzko informaziorik ematen. Horregatik, ez zen kontuan hartu.

1. Balio propioak: balio propioez osatutako 11 luzeerako bektorea (Aldagaiak 11 zirenez, eta aldagai bakoitzeko balio-propio bat zegoenez, horregatik luzeera 11 zen). Guztira 52 bektore.
2. Koordenatu nagusiak: 11×2 dimentsioko matrizea. Guztira 52 matrize (Bi zutabe hartu ziren, balio propiorik altuenak zituztenen koordenatu nagusiak lehen bi zutabeetan zeudelako).

4. Procrustes Analisia egin:
   Procrustes Analisia egiteko, koordenatu nagusiez osatutako matrizeak erabili ziren. Procrustes Analisitik, Procrustes estatistiko koefizienteez osatutako 52×52 matrizea lortu zen (Kasu bakoitza besteekiko zeukan koefizientea kalkulatu zelako). Matrize hori, hurrengo bi prozedurak egiteko erabili zen:
   - Ikasketa ez-gainbegiratua edo clustering.
   - Ikasketa gainbegiratua.
5. Datu-multzoaren gainean ikasketa ez-gainbegiratutako algoritmo hierarkiko aglomeratua eta K-means algoritmoak aplikatu.
6. Ikasketa gainbegiratutako sailkatzailea kasu normalen datuen gainean eraiki eta sailkapenaren ontasuna ikusi.

Eskema orokorra azaldu ondoren, metodo bakoitza R studion nola egin zen xehetasunekin azalduko da:

5.3.1 Dimentsio anitzeko mailaketa

Lehen urratsa, dimentsio anitzeko mailaketa izan zen. Izan ere, datuak aztertzeko eta datuen arteko antzekotasunak irudikatzeko teknika baliogarria zen. Azterketa hau, datu multzoaren korrelazio koefizienteen matrizea erabiliz sortu zen D distantzia karratuen matrize batetik hasi zen. Izan ere, D matrize hori aldagai ortogonalez, $y_1, ..., y_p$, osatutako multzo baten bidez irudikatzea zen helburua. Aldagai ortogonal horiek koordenatu nagusiak ziren.

Koordenatu nagusi horien irudikapena lortzeko, 5.7 eta 5.8 irudietan agertzen den prozedura jarraitu zen:

```
#AIRERATZEA DATU NORMALAK
p<-25
i<-1
while(i<=5){
  airfitx<-read.table(file = paste("C:/Users/Isabel/Desktop/GrAL/DatuNormalak/a
  bihamabiAldagaiak <- airfitx[,2:12]
  R <- cor(bihamabiAldagaiak) #Tarte bakoitzaren korrelazio matrizea
  D <- sqrt(1-R) #Tarte bakoitzeko distantzia matrizea
  out <-cmdscale(D, eig=TRUE, k=10) #Dimentsio anitzeko mailaketa
  if(i==1){
    plot(out$eig, type="b")
  }else{
    if(i==2){
      points(out$eig,type="b", col="red")
    }else if(i==3){
      points(out$eig,type="b", col="green")
    }else if(i==4){
      points(out$eig,type="b", col="blue")
    }else if(i==5){
      points(out$eig,type="b", col="yellow")
    }
  }
```

5.7 Irudia: Datu normalekin dimentsio anitzeko mailaketa teknika aplikatzeko urratsak.

```
  #Koordenatu nagusiak fitxategian idazteko:
  Y2 <- out$points[, 1:2]
  iz<-c("aireratzea","aireratzea","aireratzea","aireratzea","aireratzea","airerat
  pot<-c(p,p,p,p,p,p,p,p,p,p,p)
  errep<- c(i,i,i,i,i,i,i,i,i,i,i)
  konf<-data.frame(Y2,iz,pot,errep)
  write.table(konf, paste("C:/Users/Isabel/Desktop/GrAL/konfigurazioak_.txt", sep

  #Balio propioak fitxategian idazteko:
  a<-out$eig
  etiketa<- "aireratzea"
  etiketa2<- p
  etiketa3<- i
  dena<-c(a,etiketa,etiketa2,etiketa3)
  write.table(matrix(dena,nrow=1), paste("C:/Users/Isabel/Desktop/GrAL/baliopropi
  #Hurrengo errepikapenak eta potentzia ezberdinei dagozkien fitxategien datuak h
  i<-i+1
  if(i>5 && p<100){
    p<-p+25
    i<-1
  }
}
```

5.8 Irudia: Datu normalekin dimentsio anitzeko mailaketa teknika aplikatzeko urratsak.

5.8 irudia , 5.7 irudiaren jarraipena da. 5.7 eta 5.8 irudietan agertzen den zenbaki bakoitzaren azalpena emango da:

1. *p* eta *i* aldagaiak:
   5.2 atalean azaldu bezala, fitxategiek izendapen hau zeukaten: *egoera***P_E**.txt, non $\mathbf{P} = \{25, 50, 75, 100\}$ (motorraren potentzia) den eta $\mathbf{E} = \{1, 2, 3, 4, 5\}$ (errepikapenak) den. Hori dela eta, R studion *p* eta *i* aldagaiak definitu ziren, *p* potentziak adierazteko eta *i* errepikapenak adierazteko. Horri esker bakoitzari zegokion fitxategia atzitu zitekeen.

2. While bat egin zen, potentzia bakoitzaren errepikapen guztiekin prozedura berdina jarraitu behar zelako. Lehenengo, fitxategia irakurri, *read.table* komandoa erabiliz. Ondoren, korrelazio-matrizea sortu, *cor* funtzioa erabiliz. Emaitza *R* aldagaian gorde zen. Jarraitzeko, distantzia-matrizea sortu, jakinda $d_{ij} = \sqrt{1 - r_{ij}}$ zela. Behin *D* distantzia-matrizea lortu ondoren, dimentsio anitzeko mailaketa teknika aplikatu *cmdscale* funtzioa erabiliz. Dimentsio anitzeko mailaketaren emaitzak *out* aldagaian gorde ziren.

3. Irudikapena:
   cmdscale funtzioak, irteeran aldagai ezberdinez osatutako lista bat itzultzen zuen (4.2.2 atalean azalduta dago). *out* aldagaian, *cmdscale* funtzioaren emaitzak gordeta zeudenez, *out*$*eig* eginez, koordenatu nagusiak lortzeko beharrezkoak ziren *Q* antzekotasun matrizearen balio propioak atzitu zitezkeen (3.2.2 atalean, azalpena xehetasun handiagoekin ikus daiteke). Hori dela eta, *plot()* eta *points()* funtzioak erabiliz, *Q* matrizearen balio propioak (*out$eig)* irudikatu ziren zerbait esanguratsua ikusteko helburuarekin.

4. Koordenatu nagusiak:
   Pausu hau, *while*-aren barruan idatzita zegoen. Potentzia eta errepikapen bakoitzeko lortzen ziren koordenatu nagusiak, fitxategi berdinean gordetzen ziren, aplikatuko ziren beste metodo batzuetan koordenatu hauek erabili ahal izateko. Nahiz eta koordenatu nagusien dimentsioak 11×10 izan, fitxategian bakarrik 11×2 gorde ziren. Koordenatu nagusiei etiketa bat gehitu zitzaien, zein egoera (normala, helizean desoreka etab), potentzia eta errepikapenari zegokion jakiteko. *iz*, *pot* eta *errep* aldagaiak etiketak dira, eta *data.frame* erabiliz, koordenatu nagusiei etiketa horiek gehitu zitzaizkien.

```
0.377694349136184 0.00240171448595035 "aireratzea" 25 1
-0.989656016269226 -0.00361032528755046 "aireratzea" 25 1
0.304901793108399 -0.00139738157521657 "aireratzea" 25 1
0.358201654164518 -0.00631074098376998 "aireratzea" 25 1
-0.311467626379731 0.0574529686256821 "aireratzea" 25 1
-0.310569986181511 0.591783279997471 "aireratzea" 25 1
-0.305748410351367 -0.643251018097161 "aireratzea" 25 1
0.375768034754832 0.00197720052452916 "aireratzea" 25 1
0.370688608991675 0.00195739019135032 "aireratzea" 25 1
0.377694349138155 0.0024017143783462 "aireratzea" 25 1
-0.24750675011193 -0.00340480225963071 "aireratzea" 25 1
```

5.9 Irudia: Koordenatu nagusiak gordeta dituen fitxategiaren egitura. Lehenengo bi zutabeek koordenatu nagusien balioak adierazten dituzte, eta gainontzeko hiru zutabeek, etiketak. Adibide hau, datu normalen %25 potentzia duen 1 errepikapenaren datuen koordenatu nagusiak dira.

5. Balio propioak:
   Pausu hau, *while*-aren barruan idatzita zegoen. Potentzia eta errepikapen bakoitzeko lortzen ziren balio propioak, fitxategi berdinean gordetzen ziren, aplikatuko ziren beste metodo batzuetan balio hauek erabili ahal izateko. Balio propioei etiketa bat gehitu zitzaien, zein potentzia eta errepikapenari zegokion jakiteko. *etiketa*, *etiketa1* eta *etiketa2* aldagaiak, etiketak dira. *c* erabiliz, balio propioei etiketa horiek gehitu zitzaizkien.

6. *if* baldintza:
   Hasieran, $p = 25$ zen eta $i = 1$. Potentzia bakoitzeko 5 errepikapen zeudenez, i buelta bakoitzean gehitzen zen. Baina $i > 5$ izatera iristen zenean, %25 potentziari zegokion errepikapenek fitxategi guztiak irakurrita zeuden, eta hurrengo potentziaren fitxategiak irakurri ahal izateko, baldintza hori jarri zen. Horrela, $p = 50$ izatera pasako zen eta $i = 1$ izatera pasako zen.

Datu anomaloen balio propioak irudikatzeko, koordenatu nagusiak fitxategi berdinean gordetzeko eta balio propioak fitxategi berdinean gordetzeko jarraitu zen prozedura aurrekoaren berdina izan zenez, datu anomaloekiko dimentsio anitzeko mailaketa teknika aplikatzeko jarraitu ziren urratsak azaltzea ez da beharrezkoa izango.

5.3.2 Procrustes

Procrustes analisia erabat aproposa zen konfigurazioen bitartez irudikatutako bi egituren itxuren desberdintasunak aztertzeko. **Aireratzeari** zegokionez, **52** kasu zeuden (**20** kasu datu normalekin eta **32** kasu datu anomaloekin). Kasu bakoitzak, koordenatu nagusiez

osatutako 11×2 dimentsioko matrizea zeukan (5.3.1 atalean lortu zirenak). Kasu bakoitzaren matrizearekin, beste kasu guztien matrizearekiko Procrustes analisia egin zen. Izan ere, kasu bakoitzak gainontzeko 51 kasuekiko zeukan Procrustes estatistikoaren koefizientea lortzea zen helburua. Hortaz, lortu zen emaitza 52×52 matrize bat izan zen, Procrustes estatistikoaren koefizienteez osatua zegoena.

5.10 eta 5.11 irudietan, R studio erabiliz Procrustes estatistikoaren koefizientea lortzeko eman ziren pausuak azalduko dira:

```
#PROCRUSTES
konf<- read.table(file = paste("C:/Users/Isabel/Desktop/GrAL/EMAITZAKaireratzea/AIRERAT
fitxlerrokop<-length(konf[,1])
lerrokop<-c(1:fitxlerrokop)
konf<-cbind(konf,lerrokop)
zenbat<-length(konf[,1])/11 #zenbat proba dauden guztira aireratzean
lerroa<-1
zutabea<-1
bektore<-c(1:zenbat)
i<-1
j<-11
while(lerroa<=zenbat){
  i2<-1
  j2<-11
  aukeratu<-konf[i:j,1:2]
  while(zutabea<=zenbat){
    if(i2==konf[i:j,6][1]){
      bektore[zutabea]<-NA
      i2=i2+11
      j2<-j2+11
      zutabea<-zutabea+1
    }
    bestea<-konf[i2:j2,1:2]
    pp<-vegan::procrustes(aukeratu,bestea)
    ss<-pp[3]
    unlist_ss<-unlist(ss)
    bektore[zutabea]<-ss
    i2<-i2+11
    j2<-j2+11
    zutabea<-zutabea+1
    if(zutabea==zenbat){
      i2<-i2-11
      j2<-j2-11
    }
  }
```

5.10 Irudia: Procrustes Analisiaren urratsak.

```
  ema<-data.frame(bektore, lerroa)
  write.table(ema, paste("C:/Users/Isabel/Desktop/GrAL/procrustes.txt", sep = ""),
  i<-i+11
  j<-j+11
  zutabea<-1
  lerroa<-lerroa+1
}
```

5.11 Irudia: Procrustes Analisiaren urratsak.

5.11 irudia, 5.10 irudiaren jarraipena da. 5.10 eta 5.11 irudietan agertzen den kodea hitzez azalduko da:

1. Hasierako aldagaiak:
   - *konf*: kasu guztien koordenatu nagusiez osatutako 11×2 matrizeak zeuzkan fitxategiaren datuak atzitzeko balio zuen aldagaia.
   - *fitxlerrokop*: fitxategiaren lerro kopurua gordeta zeukan aldagaia.
   - *lerrokop*: bektore bat zen, $(1, ..., fitxlerrokop)$ zenbakiez osatua. *konf* aldagaiari zutabe bezala gehitzeko balio zuen.
   - *zenbat*: aireratzean zenbat kasu zeuden gordetzen zituen aldagaia. *fitxlerrokop*/11 egin zen, kasu bakoitzari fitxategiaren 11 lerro zegokiolako, koordenatu nagusiez osatutako matrizeak 11 lerro zituelako.
   - *lerroa* eta *zutabea*: indizeak ziren. Zein lerro eta zein zutabe atzitu behar ziren jakiteko.
   - *bektorea*: bektore bat zen, 52 luzerakoa. Kasu bakoitzari (besteekiko) zegokion Procrustes estatistikoaren koefizienteak gordetzen zituen.
   - *i* eta *j*: translaziorik, mailaketarik eta biraketarik jasoko ez duen matrizea hautatzeko beharrezkoak ziren indizeak. Unean analizatzen ari zen kasuaren matrizea hautatzeko balio zuten aldagai hauek.
2. Lehenengo *while*:
   52 aldiz prozesua errepikatu behar zenez, while hau sortu zen, 52 aldiz prozesua errepikatuko zela bermatzeko. Buelta bakoitzean, unean analizatzen ari zen kasua bakarrik kontuan hartzen zen. *aukeratu* aldagaian, unean analizatzen ari zen kasuaren matrizea gordetzen zen.

3. Bigarren *while*:
   i2 eta *j2* aldagaiak, gainontzeko kasuen matrizea aukeratzeko balio zuen. i, j eta $i2, j2$ indizeek bat egiten bazuten, unekoaren matrizea berriz atzitzen ari zela esan nahi zuen. Orduan *bektorea* aldagaian *NA* jartzen zen, unekoak bere buruarekiko ez zuelako Procrustes analisirik egingo. Indizeek bat egiten ez zutenean, *bestea* aldagaian, bestearen matrizea gordetzen zen. *pp* aldagaian, uneko matrizearen eta beste matrizearen Procrustes analisiaren emaitza gordetzen zen. *ss* aldagaian, Procrustes analisitik lortutako Procrustes estatistiko koefizientea gordetzen zen, eta *bektore* aldagaian emaitza ori gordetzen zen.

4. Azkenik, *bektore* aldagaian lortutako emaitza fitxategi batean gordetzen zen *write.table* aginduari esker, eta aldagaiak eguneratzen ziren.

5.3.3 Clustering

Clustergin aplikatzeko, 5.3.2 prozesuan lortutako matrizea erabili zen. Hau da, Procrustes estatistiko koefizienteez osatutako 52×52 matrizea. Matrize hori distantzia-matrizetzat erabili zen.

Hurrengo irudian, R studio erabiliz Clustering algoritmoei esker lortutako sailkapena lortzeko eman ziren urratsak azalduko dira:

```
#CLUSTERING
d<-read.table(file = paste("C:/Users/Isabel/Desktop/GrAL/EMAITZAKaireratzea/dis
etiketak<-d[,53:55]
d<-d[,1:52]
ddist <- as.dist(d)
library(cluster)
hc<-hclust(as.dist(d),method="ward.D2")
plot(hc)
K <- 10
s <- rep(NA, K)
for (k in 2:K)
{
  aux <- pam(ddist, k)$clustering
 s[k] <- mean(silhouette(aux, ddist)[,"sil_width"])
}
plot(s)
out<-pam(as.dist(d), k=4)
outC<-c(out$clustering)
plot(outD$points)
for(i in 1:length(outC)){
  if(outC[i]==1){
    points(outD$points[i,1], outD$points[i,2], col="green")
  }else if(outC[i]==2){
    points(outD$points[i,1], outD$points[i,2],col="blue")
  }else if(outC[i]==3){
    points(outD$points[i,1], outD$points[i,2],col="red")
  }else if(outC[i]==4){
    points(outD$points[i,1], outD$points[i,2], col="black")
  }
}
```

1
2
3

5.12 Irudia: Clustering metodoaren urratsak

1. Procrustes estatistiko koefizienteez osatutako 52×52 matrizea zeukan fitxategia *d* aldagaian gorde zen. Ondoren, *as.dist()* funtzioa erabili zen balio horiekin distantzia-matrizea sortzeko. Azkenik, *hclust* funtzioa erabili zen datuak sailkatzeko, eta sailkapen hori dendograma baten bidez adierazi zen. Prozedura hau, algoritmo hierarkiko aglomeratuari zegokion.

2. Oraingoan, sailkapena egiteko, $K-means$ algoritmoa aplikatu zen. Horretarako, *pam* eta *silhouette* funtzioak erabili ziren. Helburua, datu-multzorako sailkapen onena (datuak banatzeko klase kopururik onena) zein zen jakitea zen. *pam* funtzioak, klase kopuru bat emanda, kopuru horren arabera datuak sailkatzen zituen. *silhouette* funtzioak, datu-multzorako cluster (talde, *k*) optimoena zein zen identifikatu zuen. Ale bakoitzaren siluetaren batezbestekoaren balioa erabili zen cluster zenbaki optimoa jakiteko.

   Hori dela eta, *for* bat egin zen, *K* (Cluster) bakoitzaren sailkapenaren batezbes-

tekoaren balioa gordez. Adibidez $k = 2$ bazen, *pam* funtzioak k horretarako aleen sailkapena egiten zuen, hau da, datuak bi taldeetan banatzen zituen. Jarraian, datuak bi talde horietan banatuta egon ondoren, *silhouette* erabili zen. Ale bakoitzak zeukan siluetaren balioa itzultzen zuen emaitza gisa. Ale guztien siluetaren balioen batezbestekoa egin ondoren, balio hori s bektorean gordetzen zen (bektorearen indizeak cluster zenbakia adierazten zuen. Adibidez $k = 2$ clusterraren batezbestekoaren balioa bektorearen 2. posizioan gorde zen). Horri esker, amaieran bektoreko zein posizioak zeukan batezbesteko baliorik altuena ikusi ahal izan zen. Cluster hoberena, bektorean baliorik altuena zeukana izan zen. Kasu honetan cluster hoberena $k = 4$ izan zen.

3. Sailkapena jakin ondoren, puntuak grafiko batean irudikatu ziren, eta puntuak koloreekin adierazi ziren taldeak zeintzuk ziren hobeto ikusteko. Irudikatu ziren puntuak, Procrustes estatistiko koefizienteez osatutako 52×52 distanzia-matrizearekin dimentsio anitzeko mailaketa aplikatu ondoren lortzen ziren koordenatu nagusiez osatutako matrizearen lehen bi zutabeen puntuak ziren.

5.3.4 Ikasketa gainbegiratuaren sailkapena eta sailkapen horren baliozta-tzea

Teknika hau aplikatzeko, Procrustes analisitik lortu zen Procrustes estatistikoaren koefizienteez osatutako 52×52 distantzia-matrizea erabili zen.

Sailkatzailea, kasu normalen gainean bakarrik eraiki zen, *konfiantza elipsoidea*ren formula erabiliz.

52 kasuez osatutako datu-multzoa genuen, 20 kasu normalei dagokionez eta 32 kasu anomaloei dagokionez. Datu-multzoa, bi multzotan banatu behar zen: entrenamendu multzoa eta test multzoa. Kontuan izanik, erabiliko zen teknika leave-one-out zela:

- Entrenamendu multzoa: Kasu normalei zegozkien $20 - x$ kasuez osatua zegoen. x kasua, 20 horietako kasu normal bat zen, baina aldi bakoitzean, datu normal batek ezin zuen entrenamenduan parte hartu, eta, x, aldi bakoitzean entrenamendu fasean parte hartzen ez zuen kasu normala zen.

- Test multzoa: Kasu anomalo guztiek eta kasu normal guztiek osatzen zuten. Unean

entrenamenduan parte hartzen ez zuen x kasu normalak, testean bai hartzen zuen parte, gainontzeko kasu normalekin batera.

Buelta bakoitzean entrenamenduan parte hartzen ez zuen kasu normalaren gainean,x, hurrengo formula aplikatzen zen:

$$x = \frac{1}{2}\Lambda^{-1}X'(b - d^{(2)})$$

Xehetasun gehiago 3.2.2 atalean daude. x kalkulatzeko erabiltzen ziren balioak, Procrustes estatistikoaren koefizienteez osatutako 52×52 distantzia-matrizearen gainean aplikatu zen dimentsio anitzeko mailaketaren prozesuan lortutako emaitzak ziren. $x$ren balio berriarekin, konfiantza elipsoidearen sailkatzailearen formula aplikatzen zen. Horrela, sailkatzaileak ale hori normalean edo anomaloan sailkatuko zuen ikusi ahal izan zen. 5.13 irudian, prozesu horren urratsak ikus daitezke:

```
dat<-read.table(file = paste("C:/Users/Isabel/Desktop/GrAL/EMAITZAKaireratzea/disprocru
d<-as.dist(dat[,1:52])
etiketa1<-dat[,53]
etiketa2<-dat[,54]
etiketa.errep<-dat[,55]
traind<-as.matrix(d)[1:20,1:20] #normalak normalekin
testd<-as.matrix(d)[21:52,1:20]  #anomaloak normalekin
proj<-array(NA,dim=c(33,2,20))
for(i in 1:20){
  di<-traind[-i,-i]
  out<-cmdscale(di, x.ret=TRUE, eig=TRUE, k=2)
  b<-diag(-1/2*(out$x))
  Lmenos1<-diag(1/out$eig[1:2])
  X<-out$points[,1:2]
  d0<-traind[i,-i]^2 #Zergatik ez da kontuan hartzen begiratzen ari garena?
  proj[1,,i]<-1/2*Lmenos1%*%t(X)%*%matrix(b-d0,ncol=1)
  for(j in 1:32){
    d0<-testd[j,-i]^2
    proj[j+1,,i]<-1/2*Lmenos1%*%t(X)%*%matrix(b-d0,ncol=1)
  }
}
dellip<-function(x,zentroa,kobminus1){
  matrix((x-zentroa),nrow=1)%*%kobminus1%*%matrix((x-zentroa),ncol=1)
}
apply(proj[,,1],1,dellip,zentroa=c(0,0),kobminus1=solve(cov(X)))
qchisq(0.95,df=2)
ilerak<-c("i.LOO",paste("anomalo",1:32,sep=""))
zut<-paste(1:20,"normal.LOO",sep="")
emaitzak<-matrix(NA,nrow=33,ncol=20,dimnames=list(ilerak,zut))
for(i in 1:20){
  emaitzak[,i]<-apply(proj[,,i],1,dellip,zentroa=c(0,0),kobminus1=solve(cov(X)))
}
emaitzak2<-emaitzak < qchisq(0.95,df=2)
mode(emaitzak2)<-"numeric"
```

5.13 Irudia: Sailkapena eta sailkapenaren gainean egindako balioztatze prozedura.

6. KAPITULUA

Emaitzak

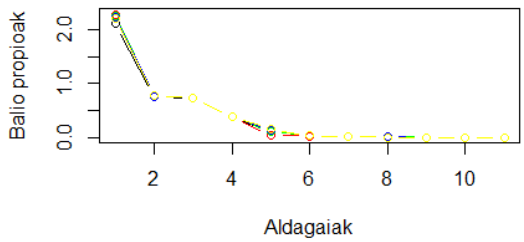
Eranskin honetan, dronaren motorrean egoera ez-normalen detekziorako lortu ziren emaitzak azalduko dira. 5 atalean azaldu den bezala, egoera bakoitzeko datuen analisia egin zenez, egoera bakoitzerako lortu ziren emaitzak azalduko dira.

6.1 Aireratzea

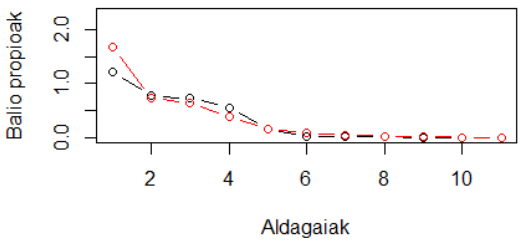
Aireratze egoeran lortutako emaitzak:

6.1.1 Balio propioak

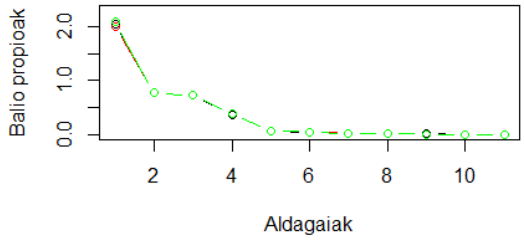
Potentzia bakoitzerako, datu normalen zein datu anomaloen balio propioen irudikapena egin zen. Ez da beharrezkoa potentzia guztien balio propioak irudikatzean. Adibidez, motorraren potentzia %25 zenean, jaso ziren datu-normalen zein anomaloen balio propioen irudikapena hurrengoa da:



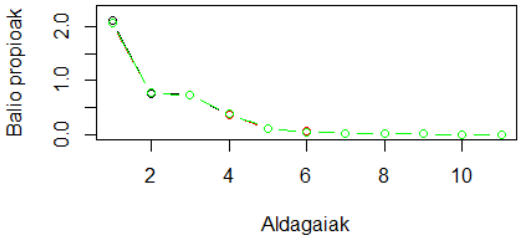
(a) Datu normalak.



(b) Datu anomaloak: Desoreka.



(c) Datu anomaloak: %5 zarata.



(d) Datu anomaloak: %10 zarata.

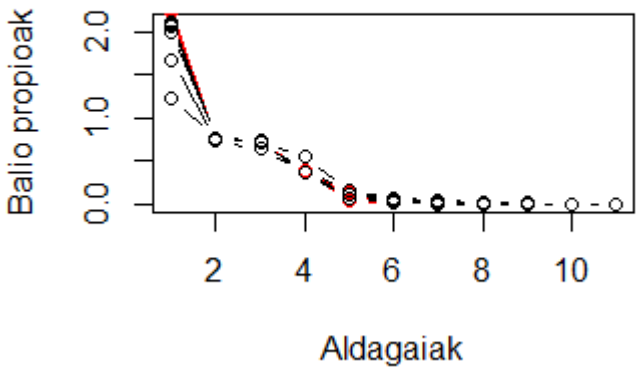
6.1 Irudia: Aireratze egoeraren dronaren motorraren %25eko potentziarekin jasotako datuen balio propioak.

Grafikoetan agertzen ziren koloreak, zein errepikapen ziren adierazten zuten:

Kolorea	Errepikapena
	1
	2
	3
	4
	5

6.1 Taula: Errepikapenak.

Balio propio guzti horiek irudi bakar batean elkartzean, hurrengo irudikapena agertu zen:



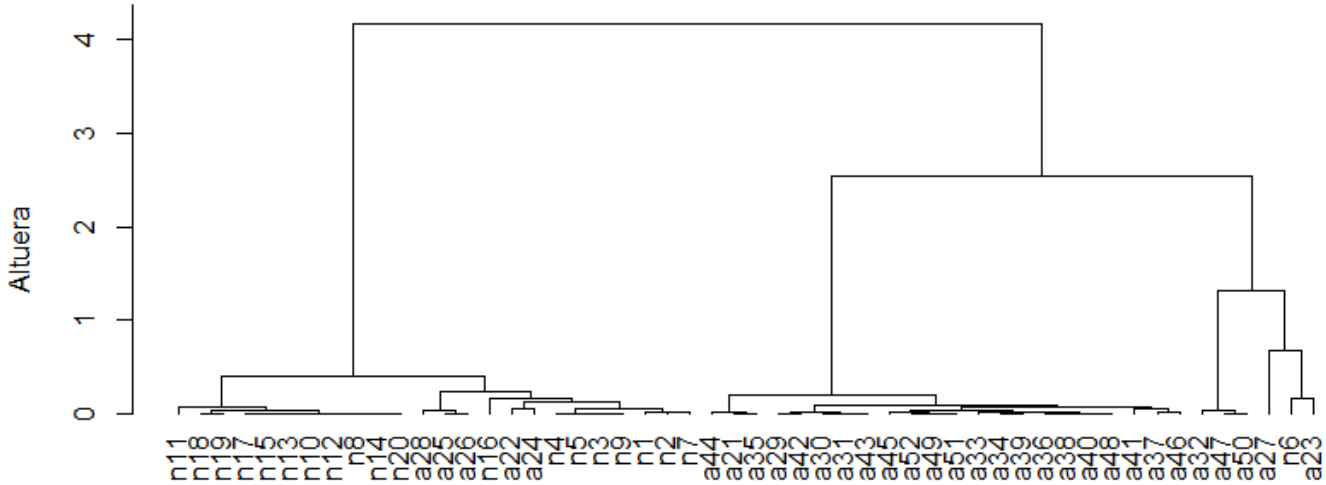
6.2 Irudia: Aireratze egoeraren %25 potentziaren balio propio guztiak irudi bakar batean

Kolorea	Datu-mota
	normala
	anomaloa

6.2 Taula: Aireratzen egoeraren %25potentziaren datu normalen eta anomaloen balio propioen koloreak.

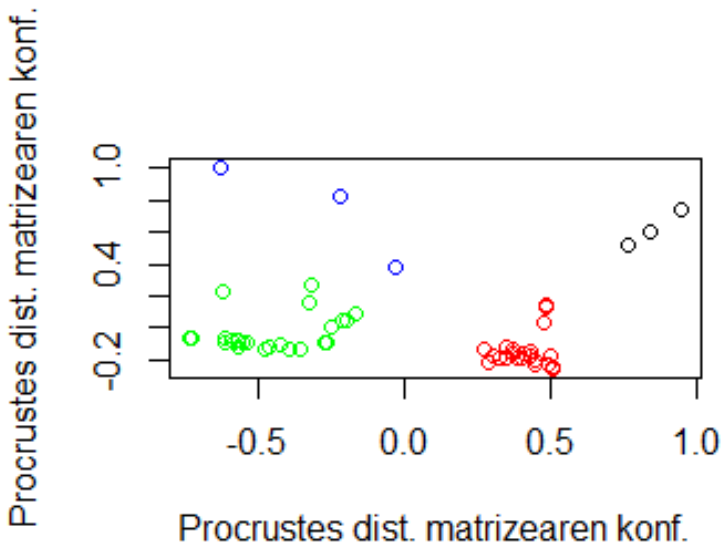
Datu normalen eta anomaloen balio propioak bi koloreetan irudikatu ziren elkarren artean bereiztu ahal izateko. Baina, puntu guztiak kolore berdinarekin irudikatu izan balira, datu normalen balio propioak ez ziren datu anomaloetaz bereiziko. Hau da, balio propioen irudikapenak ez zuen datu normalen eta anomaloen arteko sailkapena egiten lagundu, bereizi ez zirelako.

6.1.2 Clustering



6.3 Irudia: Aireratzearen sailkapena erakusten duen dendrograma.

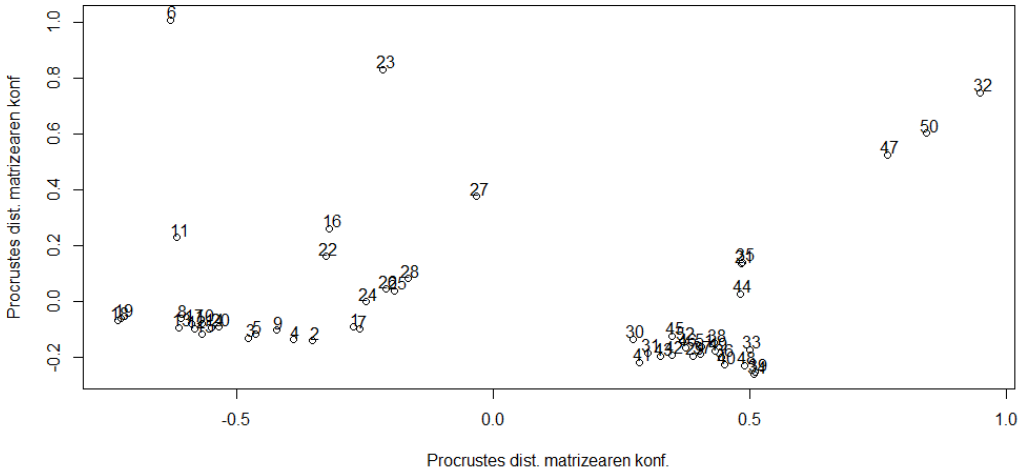
Dendrograma hau, 52×52 dimentsioko Procrustes estatistiko koefizienteez osatutako distantzia-matrizea erabiliz sortu zen. Dendrogramari begiratuz, mozketa 1 luzeran egiten bazen, lau talde bereizten zirela ikus zitekeen. Aurretik **"a"** idatzita zeukaten zenbakiek, kasu anomaloak ziren, eta aurretik **"n"** idatzita zeukatenak aldiz, kasu normalak. Clustering aplikatuz, datuak normaletan eta ez-normaletan bereizi zirela ikusi zitekeen. Dendrograman zenbakiak oso ondo bereizten ez zirenez, puntu bakoitzaren irudikapena egin zen, kasu bakoitza zein multzotan zegoen hobeto ikusteko.



6.4 Irudia: Aireratzea egoerari zegokion datuen sailkapena taldeka.

Irudikapen honekin, dendrograman bereizten zen bezala, lau talde zeuden. Talde bakoitza zein kasuez osatua zegoen azalduko da:

- Talde **urdina**: datu normalez zein helizean desoreka jasan zuten datu anomaloez osatua zegoen
- Talde **berdea**: datu normalez zein helizean desoreka jasan zuten datu anomaloez osatua zegoen
- Talde **gorria**: helizean desoreka jasan zuten datu anomaloez eta azelerazio aldagaian %5eko eta %10eko zarata jasan zuten datu anomaloez osatua zegoen
- Talde **beltza**: azelerazio aldagaian %5eko eta %10eko zarata jasan zuten datu anomaloez osatua zegoen



6.5 Irudia: Aireratzea egoerari zegokion datuen sailkapena taldeka.

Irudi honekin, talde bakoitzaren kideak zeintzuk ziren ikus zitekeen. Orokorrean, datu anomaloak anomaloekin batu ziren eta datu normalak normalekin, bi taldetan izan ezik, urdina eta berdea. Horregatik, talde horretako kideak zeintzuk ziren zehaztuko da:

Talde urdinari dagokionez, 6., 23., eta 27. kasuek osatzen zuten.

- 6.kasua: **datu normala->%50 potentzia->1 errepikapena** da.
- 23.kasua: **datu anomaloa -> desoreka -> %50 potentzia -> 1 errepikapena** da.
- 27.kasua: **datu anomaloa->desoreka->%100 potentzia -> 1 errepikapena** da.

Talde berdeari dagokionez, (1, ..., 5), (7, ..., 20) datu normalen kasuek eta (22,24,25,26,28) datu anomaloen kasuek osatzen zuten.

- Datu normal guztiak zeuden, talde urdinean zegoena 6. kasua izan ezik.
- Helizean desoreka jasan zuten datu anomaloez osatua zegoen.

Nahiz eta talde guztiak guztiz homogeneoak ez izan, hori ikusita, Procrustes distantziek datuak normaletan eta ez-normaletan bereizteko gaitasuna zutela ondoriozta zitekeen.

6.1.3 Sailkapena

	SAILKAPENA	**SAILKAPENA**	
ERREALA	**Normala**	**Ez normala**	Zeintzuk
Normala:	18 (**%90**)	2 (%10)	6, 16
Ez normala:	1 (%3,125)	31 (**%96,875**)	22

6.3 Taula: Aireratzearen sailkapena ikasketa gainbegiratua eginez.

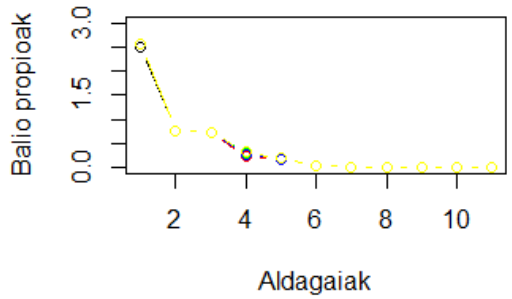
6. kasua %50 potentziako 1 errepikapenari zegokion datu normala zen. 16.kasua, %100 potentziako 1 errepikapenari zegokion. Nahiz eta bi datuek, datu normal bezala sailkatua egon, datu anomalotzat hartu ziren. Hori, 1 errepikapenean motorraren sentsoreen funtzionamendua oraindik %100ean funtzionatzen ez zeudelako izan zitekeen. %22.kasua, helizean desoreka jasan zuen %25 potentziako 2.errepikapenari zegokion.

6.2 Lurreratzea

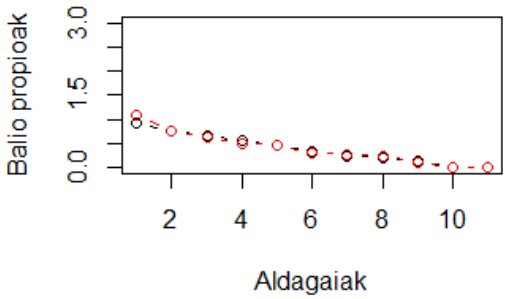
Lurreratze egoeran lortutako emaitzak:

6.2.1 Balio propioak

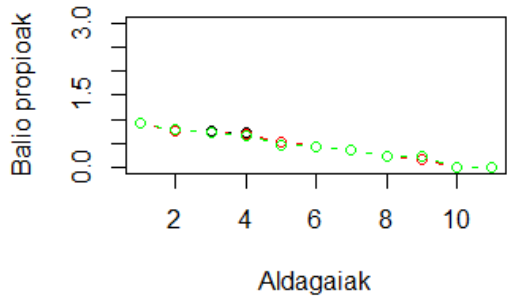
Potentzia bakoitzerako, datu normalen zein datu anomaloen balio propioen irudikapena egin zen. Potentzia guztiekin prozedura bera jarraitu zenez, ez da beharrezkoa potentzia guztien balio propioak irudikatzea. Adibidez, motorraren potentzia %25 zenean, jaso ziren datu-normalen zein anomaloen balio propioen irudikapena egingo da:



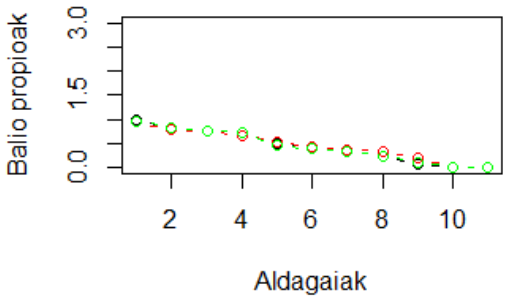
(a) Datu normalak.



(b) Datu anomaloak: Desoreka.



(c) Datu anomaloak: %5 zarata.



(d) Datu anomaloak: %10 zarata.

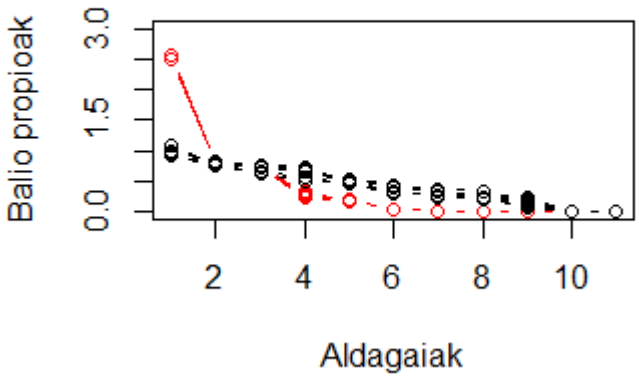
6.6 Irudia: Lurreratze egoeraren dronaren motorraren %25eko potentziarekin jasotako datuen balio propioak.

Grafikoetan agertzen ziren koloreak, zein errepikapen ziren adierazten zuten:

Kolorea	Errepikapena
	1
	2
	3
	4
	5

6.4 Taula: Errepikapenak.

Balio propio guzti horiek irudi bakar batean elkartzean, hurrengo irudikapena agertu zen:



6.7 Irudia: Lurreratze egoeraren %25 potentziaren balio propio guztiak irudi bakar batean.

Kolorea	Datu-mota
	normala
	anomaloa

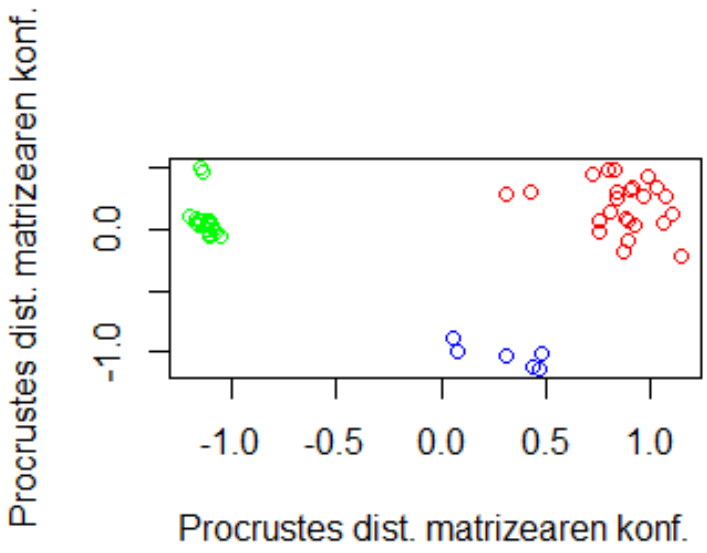
6.5 Taula: Lurreratze egoeraren %25 potentziaren datu normalen eta anomaloen balio propioen koloreak.

Bi koloretan irudikatu ziren datu normalen eta anomaloen balio propioak, bereiztu ahal izateko. Baina, puntu guztiak kolore berdinarekin irudikatu izan balira, datu normalen balio propioak ez ziren datu anomaloetaz bereiziko. Hau da, balio propioen irudikapenak ez zuen datu normalen eta anomaloen arteko bereizketan lagundu. Hala ere, lurreratze egoeran balio propioen baliorik handiena zeukatenak datu normalei zegokiela ikus daiteke. Baina, hala ere, informazio hori ez zen nahikoa balio propio altua edukitzeagatik datua normala zela eta baxua edukitzeagatik datua anomaloa zela bereizteko.

6.2.2 Clustering

6.8 Irudia: Lurreratzearen sailkapena erakusten duen dendrograma.

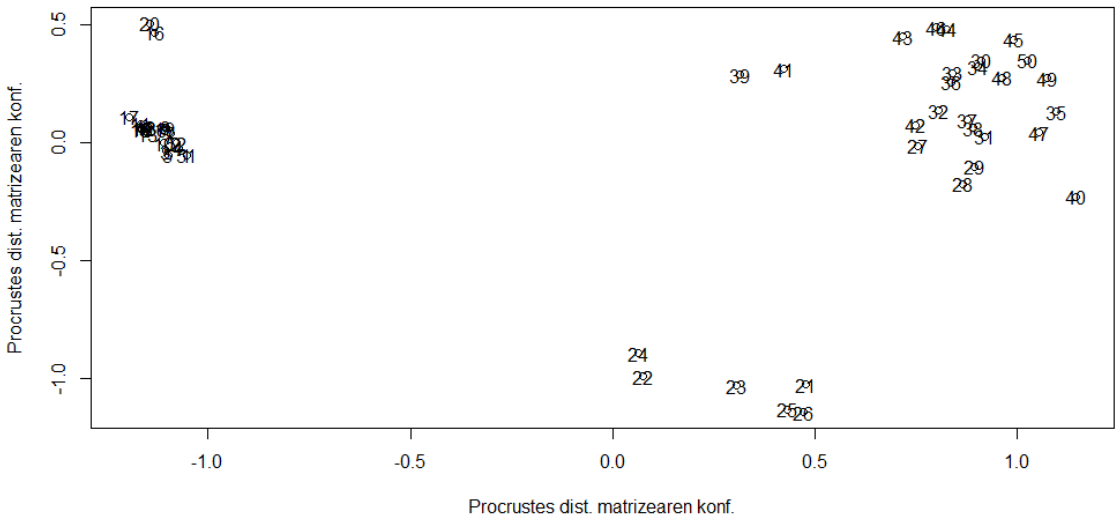
Dendrogramari erreparatuz, mozketa 2 luzeran egiten bazen, hiru talde bereizten zirela ikus zitekeen. Aurretik **"a"** idatzita zeukaten zenbakiek, kasu anomaloak ziren, eta aurretik **"n"** idatzita zeukatenak aldiz, kasu normalak. Datu normalak anomaloetatik nahiko bereizita zeudela ikus zitekeen. Bazirudien clustering egin ondoren talde berdinetan taldekatutako datu guztiak mota berdinekoak zirela. Ez zegoen normalen eta anomaloen nahasketarik talde bakar batean. Hori hobeto ikusteko, puntu bakoitzaren irudikapena egin zen, kasu bakoitza zein multzotan zegoen hobeto ikusteko.



6.9 Irudia: Lurreratze egoerari dagokion datuen sailkapena taldeka.

Irudikapen honekin, dendrograman bereizten zen bezala, hiru talde zeuden. Talde bakoitza zein kasuez osatua zegoen azalduko da:

- Talde **berdea**: datu normalez osatua zegoen.
- Talde **urdina**: helizean desoreka jasan zuten datu anomaloez osatua zegoen.
- Talde **gorria**: azelerazio aldagaian %5eko eta %10eko zarata jasan zuten datu anomaloez osatua zegoen.



6.10 Irudia: Lurreratze egoerari dagokion datuen sailkapena taldeka.

Irudi horrekin, talde bakoitzaren kideak zeintzuk ziren ikus zitekeen. Datu anomaloak anomaloekin batu ziren eta datu normalak normalekin. Beraz, datuak normaletan eta anomaloetan bereiztea posiblea zela ondoriozta zitekeen.

6.2.3 Sailkapena

	SAILKAPENA	**SAILKAPENA**	
ERREALA	**Normala**	**Ez normala**	Zeintzuk
Normala:	17 (**%85**)	3 (%15)	16,18,20
Ez normala:	0 (%0)	32 (**%100**)	-

6.6 Taula: Lurreratzearen sailkapena ikasketa gainbegiratua eginez.

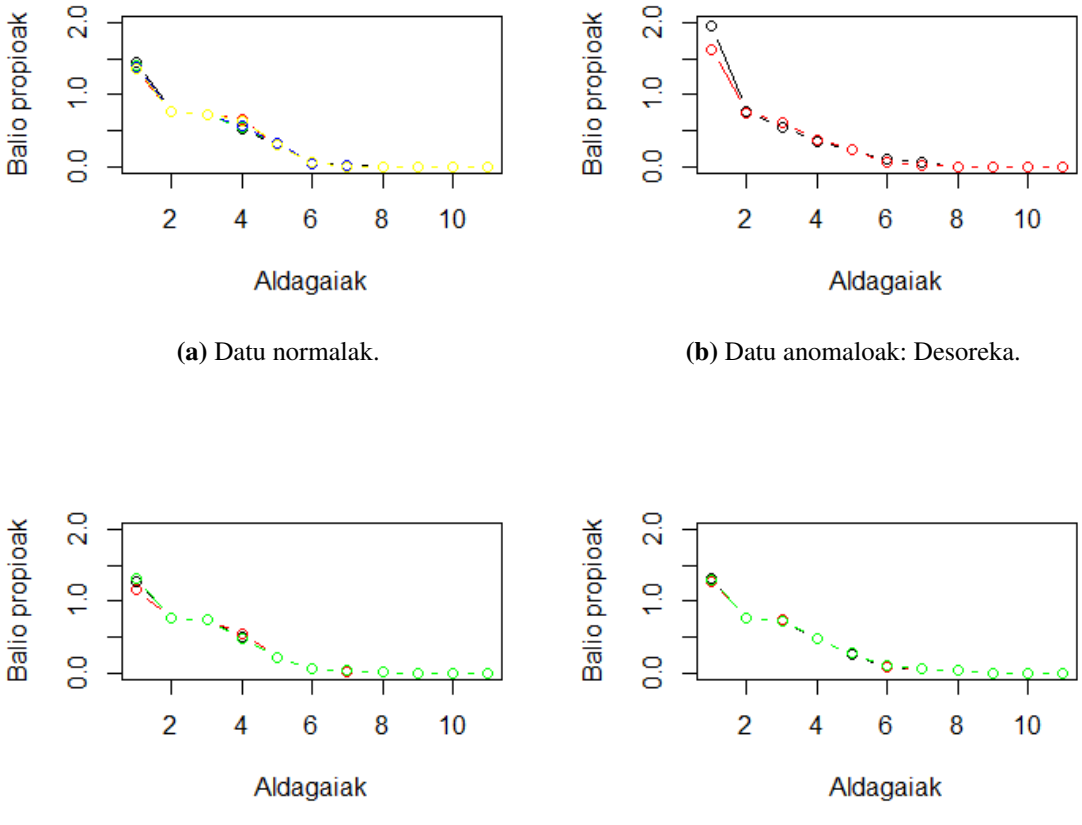
16., 18. eta 20. kasuak, %100 potentziaren 1,3 eta 5 errepikapenei zegozkien. Datu hauek normalak izanda, anomalotzat sailkatu ziren. Anomaloak ziren datuak, anomalotzat sailkatu ziren.

6.3 Hoztea

Hoztea egoeran lortutako emaitzak:

6.3.1 Balio propioak

Datu normalen zein datu anomaloen balio propioen irudikapena potentzia ezberdin bakoitzerako egin zenez, ez da beharrezkoa potentzia guztien balio propioak irudikatzea. Adibidez, motorraren potentzia %25 zenean, jaso ziren datu normalen zein anomaloen balio propioen irudikapena hurrengo irudian adieraziko da:



(a) Datu normalak.

(b) Datu anomaloak: Desoreka.

(c) Datu anomaloak: %5 zarata.

(d) Datu anomaloak: %10 zarata.

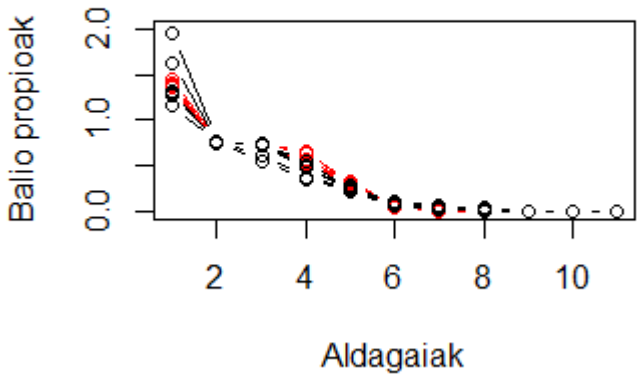
6.11 Irudia: Hoztea egoeran dronaren motorraren %25eko potentziarekin jasotako datuen balio propioak.

Grafikoetan agertzen ziren koloreak, zein errepikapen ziren adierazten zuten:

Kolorea	Errepikapena
	1
	2
	3
	4
	5

6.7 Taula: Errepikapenak.

Balio propio guzti horiek irudi bakar batean elkartzean, hurrengo irudikapena agertu zen:



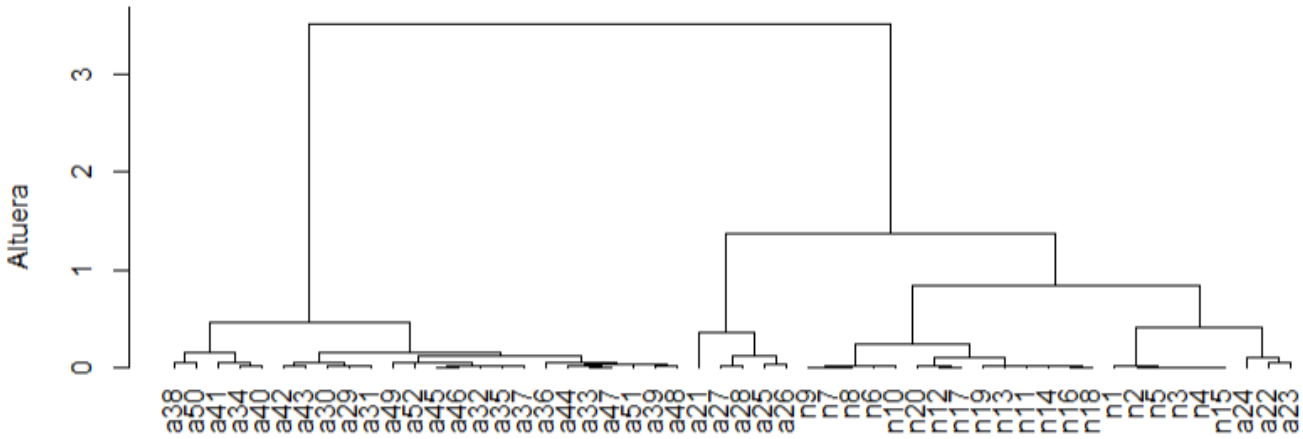
6.12 Irudia: Hoztea egoeraren %25 potentziaren balio propio guztiak irudi bakar batean.

Kolorea	Datu-mota
	normala
	anomaloa

6.8 Taula: Hoztea egoeraren %25 potentziaren datu normalen eta anomaloen balio propioen koloreak.

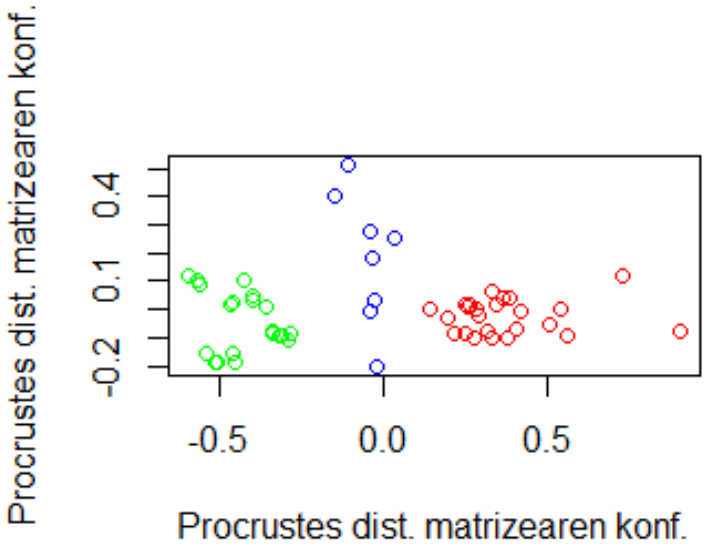
Normalen eta anomaloen balio propioak bi koloreetan irudikatu ziren, beraien artean bereiztu ahal izateko. Baina, puntu guztiak kolore berdinarekin irudikatuak izan balira, datu normalen balio propioak ez ziren datu anomaloetaz bereiziko. Hau da, balio propioen irudikapenak ez zuen datu normalen eta anomaloen bereizmenean lagundu, haien artean bereizten ez zirelako.

6.3.2 Clustering



6.13 Irudia: Hoztearen sailkapena erakusten duen dendrograma.

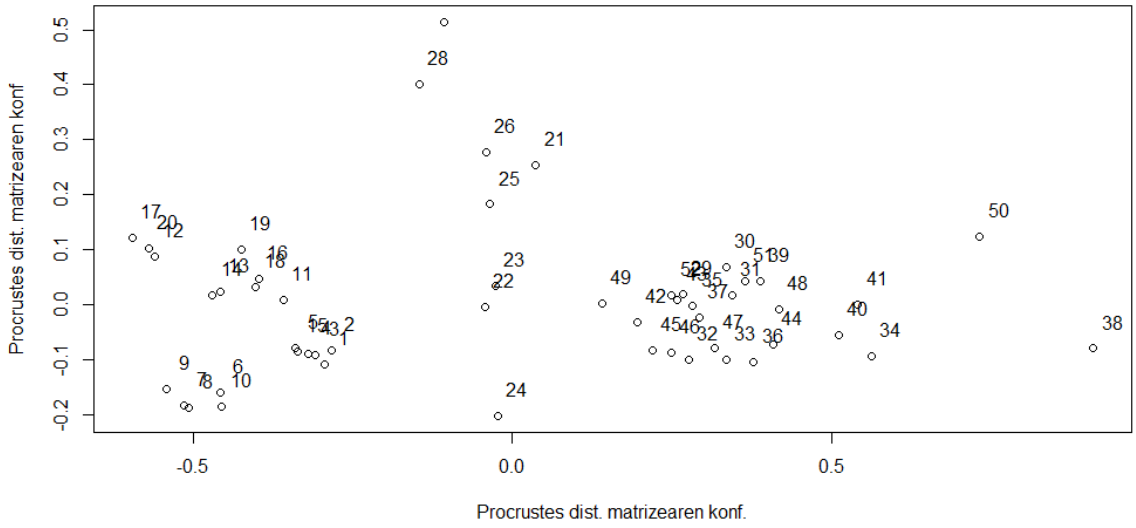
Dendrogramari erreparatuz, mozketa 1 luzeran egiten bazen, hiru talde bereizten zirela ikus zitekeen. Aurretik **"a"** idatzita zeukaten zenbakiek, kasu anomaloak ziren, eta aurretik **"n"** idatzita zeukatenak aldiz, kasu normalak. Datu normalak anomaloetatik nahiko bereizita zeudela ikus zitekeen.



6.14 Irudia: Lurreratze egoerari dagokion datuen sailkapena taldeka.

Irudikapen honekin, dendrograman bereizten zen bezala, hiru talde zeuden. Talde bakoitza zein kasuez osatua zegoen azalduko da:

- Talde **berdea**: datu normalez osatua zegoen
- Talde **urdina**: helizean desoreka jasan zuten datu anomaloez osatua zegoen
- Talde **gorria**: azelerazio aldagaian %5eko eta %10eko zarata jasan zuten datu anomaloez osatua zegoen



6.15 Irudia: Lurreratze egoerari dagokion datuen sailkapena taldeka.

Irudi honekin, talde bakoitzaren kideak zeintzuk ziren ikus zitekeen. Datu anomaloak anomaloekin batu ziren eta datu normalak normalekin. Beraz, datuak normaletan eta anomaloetan bereiztea bazegoela ondoriozta zitekeen.

6.3.3 Sailkapena

	SAILKAPENA	**SAILKAPENA**	
	Normala	**Ez normala**	Zeintzuk
Normala:	19 (**%95**)	1 (%5)	17
Ez normala:	0 (%0)	32 (**%100**)	-

6.9 Taula: Hoztearen sailkapena ikasketa gainbegiratua eginez.

17. kasua, %100eko potentzia zuen 2.errepikapenari zegokion datu normala zen. Datu normaltzat hartu eta anomalotzat sailkatu zen. Anomalotzat hartu ziren datuak, anomaloetan sailkatu ziren.

7. KAPITULUA

Ondorioak

Atal honetan, proiektuaren garapena egin eta lortutako emaitzak aztertu ondoren, proiektuari buruz ateratako ondorioak azalduko dira, bai eta ondorio pertsonalak ere. Bestalde, proiektuarekin zerikusia izan dezaketen etorkizunerako lanak ere gainetik aztertuko dira.

7.1 Proiektuaren ondorioak

Hauek izan ziren proiektua amaitzean ondorioztatu ziren ondorioak:

7.1.1 Emaitza esperimentalak

Proiektua garatzen hasi baino lehen eta datuak analizatzen hasi baino lehen, zer nolako emaitzak aterako ziren ezin zen jakin.

Proiektu hau itsuan aurrera eraman zela esan daiteke. Izan ere, gure helburu nagusia datuak normaletan edo ez-normaletan bereizteko gai izatea zen. Baina, ezerk ez zigun bermatzen proiektuaren amaieran bereizmen hori egitea posiblea izango zenik.

Beraz, proiektua itsuan hasi zen eta aurrera egin ahala gauzak ikusten hasi ziren.

Orokorrean, balio propioen irudikapenak datuak normaletan edo anomaloetan bereizteko ahaleginetan ez zuen batere lagundu. Horregatik, bide hori baztertu eta koordenatu

nagusiak erabili ziren bide berri batetik aurrera egiteko.

Bide berri horretatik lortu ziren emaitzetatik, ondorioztatu zitekeena hurrengoa da:

- Ikasketa ez-gainbegiratua erabiliz:
  - Orokorrean, hoztea eta lurreratzea egoeretan, ikasketa ez-gainbegiratua erabiliz, datu normalak eta anomaloak ondo bereizi ziren. Horregatik, sailkapenaren emaitzan, anomaloak anomaloekin taldekatu ziren eta normalak normalekin.
  - Aireratze egoeran aldiz, datuak okerrago sailkatu ziren. Ikasketa ez-gainbegiratua sailkapena egiteko erabili zenean, bi talde ezberdinetan datu normalak zein anomaloak taldekatuta zeuden.
- Ikasketa gainbegiratua erabiliz:
  - Hoztea eta lurreratze egoeratan, anomaloak ziren kasu guztiak anomaloetan sailkatu ziren. Aireratze egoeran, anomaloak ziren kasu guztiak anomaloetan sailkatu ziren bat izan ezik.
  - Hiru egoeratan kasu normala zen %100 potentziaren errepikapenen bat anomalotzat hartu zen.

7.1.2 Emaitza teknikoak

Emaitza teknikoei dagokionez, metodo matematiko eta estatistiko berri ugari ikasi ziren. Proiektu honen garapenean, ez zen software eta hardware teknologia ugarirekin lan egin. Proiektu hau praktikoa baino, teorikoagoa izan zela esan zitekeen. Hau da, eduki teoriko gehiago ikasi ziren hauek praktikan jartzeko tresnak baino.

Proiektu hau aurrera eramateko erabili zen softwarea R studio izan zen. Oso erreminta ahaltsua izan zen. Edozein kalkulu mota egitea ahalbidetzen zuen. Ehundaka datuen kalkuluak nahiko modu azkarrean kalkulatzen zituen eta grafikoak egiteko oso tresna egokia izan zen.

7.2 Etorkizunerako lana

Proiektu honekin aurrera jarraitu behar izango balitz, hurrengo pausua, datuen analisia *online* egitea izango litzateke.

Orain arte egin den datuen analisia, *offline* izan da. Hau da, aireratze egoera hasi eta amaitu egin den arte, ez da daturik analizatu. Datuak, egoeraren amaierara iristean analizatu dira. Hau da, drona piztu bezain pronto eta 5 minutu pasa ondoren aireratze egoera amaitzen bada, 5 minutu horietan zehar dronaren motorraren sentsoreek informazioa biltegiratzen egon dira, eta 5 minutuak pasa ondoren, biltegiratutako datuak atzitu eta analizatu dira.

Hori dela eta, 5. minutura iritsi baino lehen, 30 segundoro dronaren motorreko sentsoreek jasotako datuak analizatzea hurrengo pausua izango litzateke. Horrekin, lehen 30 segundoen datuak analizatuz eta datuak normaletan eta anomaloetan bereiztea lortu ondoren, hurrengo [30-T, 30+T] segundoetan jasoko ziren datuen bereizmena aurreikustea lortu nahiko zen. T, zenbaki finko bat izango litzateke, segundo finko bat, aurreko tartearen eta hurrengo tartearen datuak kontuan hartzea ahalbidetuko zuena.

7.3 Ondorio pertsonalak

Niri dagokionez, proiektu bat bakarrik egitearen zailtasunei aurre egiten ikasi nuen. Zailtasun handiena ezjakintasuna izan zen, bai oinarri teorikoei bai praktikari zegokionez.

Oinarri teorikoei zegokionez, metodo bat aplikatu ahal izateko, metodo horri buruzko informazio asko bilatu behar nuen. Batetik, kontzeptuak ondo ulertzeko, eta, bestetik, metodo hori aplikatuz lortzen ziren emaitzak ulertzen eta interpretatzen jakiteko.

Praktikari dagokionez, R studio softwareak nola funtzionatzen zuen ikasi behar izan nuen. Eta, horretarako, Interneten ere gauza mordoa bilatu eta irakurri behar izan nituen. Askotan, zerbait egin nahi nuen, eta zerbait hori R studion nola egiten zen ez nekienez, Interneten bilatu behar nuen.

Zailtasun horiek gainditu ondoren, independenteago izaten eta gauzak nire kabuz konpontzen ikasi nuen. Hala ere, proiektuaren zuzendaria beti egon zen laguntzeko prest, eta,

nola ez ba, laguntza handia ere eman zidan zailtasunak eduki nituenean.

Eranskinak

Bibliografia

[1] ALER, R. *Evaluación de técnicas de aprendizaje*. [PDF fitxategia] http://ocw.uc3m.es/ingenieria-informatica/analisis-de-datos/transparencias/evaluacion.pdf.

[2] AMAT, J. *Clustering y heatmaps: aprendizaje no supervisado*. https://rpubs.com/Joaquin_AR/310338, 2017ko iraila.

[3] ANONIMOA. *Criterios de similitud. Similitud, divergencia y distancia*. https://www.uv.es/ceaces/multivari/cluster/criterios_de_similitud.htm.

[4] ANONIMOA. *Introducción al Análisis Cluster*. https://www.uv.es/ceaces/multivari/cluster/CLUSTER2.htm.

[5] ANONIMOA. *Resumen análisis Cluster*. [PDF fitxategia]. https://www.ugr.es/ mvargas/2.RESUMENANLISISCLUSTER.pdf.

[6] CUADRAS, C.M. *Nuevos métodos de análisis multivariante*. [PDF fitxategia]. http://www.ub.edu/stat/personal/cuadras/metodos.pdf, 2019.

[7] CUADRAS, C.M. (1989). Distancias estadísticas. *Revista Estadistica Española*. Volumen 30, Número 119. Páginas 295-378.

[8] GRANÉ, A. *Distancias estadísticas y Escalado Multidimensional (Análisis de Coordenadas Principales)*. [PDF fitxategia]. http://halweb.uc3m.es/esp/Personal/personas/agrane/ficheros_docencia/MULTIVARIANT/slides_Coorp_reducido.pdf.

[9] GIL, C. *Métodos de remuestreo y validación de modelos: Validación cruzada y Bootstrap*. https://rpubs.com/Cristina_Gil/CV_Bootstrap, 2018ko maiatza.

[10] MARTÍNEZ, E. *Escalado multidimensional*. [PDF fitxategia]. http://intranetua.uantof.cl/facultades/csbasicas/Matematicas/academicos/emartinez /magister/escalado.pdf.

[11] MUÑUZURI, J. *Técnicas de clusterización*. [PDF fitxategia]. http://bibing.us.es/proyectos/abreproy/5453/fichero/PFC+tecnicas+clusterizacion.pdf, 2014ko iraila.

[12] PEREZ, I. & TORCIDA, S. (2012). Análisis de Procrustes y el estudio de la variación Morfolófica. *Revista Argentina de antropología biológica*. Volumen 14, Número 1. Páginas 131-141.

[13] RUIZ, L. *Coeficiente de correlación de Pearson: qué es y cómo se usa*. https://psicologiaymente.com/miscelanea/coeficiente-correlacion-pearson.

[14] WIKIPEDIA. *Coeficiente de correlación de Pearson*. https://es.wikipedia.org/wiki/Coeficiente_de_correlaci%C3%B3n_de_Pearson, 2019ko maiatzaren 22.

[15] WIKIPEDIA FUNDATION *Clasificadores (matemático)*. https://esacademic.com/dic.nsf/eswiki/270665#Aprendizaje_y_Miner.C3.ADa_de_datos, 2010.

[16] WIKIPEDIA. *Validación cruzada*. https://es.wikipedia.org/wiki/Validaci%C3%B3n_cruzada, 2019ko uztailaren 13.

[17] WIKIPEDIA. *Clasificador (matemáticas)*. https://es.wikipedia.org/wiki/Clasificador_(matemáticas), 2018ko uztailaren 1.

[18] WIKIPEDIA. *Distancia de Mahalanobis*. https://es.wikipedia.org/wiki/Distancia_de_Mahalanobis, 2015ko apirilaren 6.