

# Interfície Web per aprendre Aprementatge

Facultat d'Informàtica de Barcelona  
Universitat Politècnica de Catalunya - BarcelonaTech

GRAU EN ENGINYERIA INFORMÀTICA  
COMPUTACIÓ

*Autora:*  
*Sònia Sandalinas Pérez*

*Director:*  
*Luis Antonio Belanche.*  
*Departament:*  
*Ciències de la Computació*

25 de Maig del 2015

# ÍNDEX

<b>ÍNDEX</b> .....	<b>2</b>
<b>Índex de Figures:</b> .....	<b>5</b>
<b>Índex de Taules</b> .....	<b>7</b>
<b>Índex de Codi</b> .....	<b>8</b>
<b>Resums</b> .....	<b>9</b>
Català.....	9
English .....	9
Castellano .....	10
<b>PRIMERA PART: INTRODUCCIÓ I PLANIFICACIÓ</b> .....	<b>11</b>
<b>1 - Introducció</b> .....	<b>12</b>
1.1 - Estat de l'art.....	13
1.1.1 - <i>Machine Learning</i> .....	13
1.1.2- R.....	14
1.1.3 – Rstudio.....	14
1.1.4 - RShiny.....	14
1.1.5 - Aplicacions actuals .....	15
<b>2 - ABAST</b> .....	<b>17</b>
2.1 - Objectius i requeriments .....	17
2.1.1 - Objectius generals .....	17
2.1.2 - Requeriments .....	17
2.1.2.1 - Administrador / Professor :.....	18
2.1.2.2 - Usuari : .....	18
2.2 – Abast del TFG .....	19
<b>3 - Metodologies</b> .....	<b>20</b>
3.1 - Mètodes de treball .....	20
3.2 - Eines de seguiment.....	21
3.3 - Mètode de validació .....	21
3.4 - Algoritmes i programari.....	21
3.4.1 - Llenguatges de programació : .....	22
3.4.2 - Aplicació Web : .....	22
<b>4 - Planificació Temporal</b> .....	<b>23</b>
4.1 - Fases del projecte .....	23
4.1.1 - Planificació i viabilitat del projecte.....	23
4.1.2 - Repàs i estudi dels conceptes de <i>Machine Learning</i> (10/10/2014 - 19/12/2014) .....	24
4.1.3 – Recull d'informació i estudi de les noves tecnologies (19/12/2014 - 2/01/2015).....	24

4.1.4 - Disseny de l'aplicació web (02/01/2015 - 22/01/2015)	25
4.1.5 – Programació i testing de l'aplicació (22/01/2015 - 15/04/2015)	25
4.1.6 - Part final del projecte (16/04/2015 - 25/05/2015)	25
4.2 - Resum temps del projecte	25
4.3 – Diagrama de Gantt Modificat	26
<b>5 - Gestió econòmica i pressupost</b>	<b>26</b>
5.1 - Identificació dels costos	26
5.2 - Costos recursos humans	26
5.3 – Costos de Hardware	28
5.4 – Costos Software	28
5.5 – Altres despeses	29
5.6 - Variacions del pressupost inicial	29
5.7 – Pressupost total	30
<b>6 - Sostenibilitat i compromís social</b>	<b>31</b>
6.1 Valoració de la sostenibilitat del projecte	31
6.2 - Matriu de sostenibilitat	32
<b>SEGONA PART: Descripció i Implementació</b>	<b>33</b>
<b>1 - Definició de conceptes principals</b>	<b>34</b>
1.1 - Descripció del problema	34
<b>2 - Definició de les “Associated Tasks”</b>	<b>36</b>
2.1 - Classificació	36
2.1.1 - Conceptes	36
2.1.2 - Enfocament general per la resolució d'un problema de classificació	38
2.2 - Regressió	40
2.2.1 - Conceptes	41
2.2.2 - Enfocament general per la resolució d'un problema de regressió	42
2.2.3 - Eficàcia d'un model de regressió	43
2.3 - Clustering	43
2.3.1 - Conceptes	43
2.3.3 - Eficàcia:	47
2.4 - Extracció de Característiques	48
2.4.1 - Conceptes:	48
<b>3 - Disseny, Implementació i Satisfacció dels requisits</b>	<b>49</b>
3.1 – Disseny	49
3.2 – Implementació i Satisfacció de Requisits	53
3.2.1 - Administrador / Professor :	53
3.2.2 - Usuari:	57
<b>4 - Assoliment competències Tècniques</b>	<b>62</b>
<b>5 - Exemple d'execució d'aplicació</b>	<b>64</b>
5.1 – Exemple d'execució de la part de “Login”	65
5.2 - Exemple d'execució de la part de learn	65

5.3 - Exemple d'execució de la part de test .....	68
<b>TERCERA PART: MILLORES I CONCLUSIONS.....</b>	<b>72</b>
<b>1 - Millores Futures .....</b>	<b>73</b>
<b>2 - Conclusions .....</b>	<b>74</b>
<b>Bibliografia.....</b>	<b>75</b>
<b>Glossari .....</b>	<b>76</b>
<b>Annexes.....</b>	<b>77</b>

# Índex de Figures:

<i>Figura 1 - Aplicació de learn K-means.</i>	16
<i>Figura 2: Web per aprendre estadística</i>	16
<i>Figura 3: Definició gràfica de la tasca de Classificació.</i>	37
<i>Figura 4: Exemple de iris dataset.</i>	37
<i>Figura 5: Registre a classificar.</i>	38
<i>Figura 6: Resolució d'un problema de classificació.</i>	39
<i>Figura 7: Matriu de resultats d'Iris Dataset.</i>	40
<i>Figura 8: Exemple de dataset on X2 depèn de X1.</i>	41
<i>Figura 9: Registre a trobar el valor de la variable dependent "X2".</i>	42
<i>Figura 10: Resultat d'aplicar un model de regressió a una variable dependent "X2"</i>	43
<i>Figura 11: Exemple de clústers.</i>	44
<i>Figura 12: Exemple clústers en iris dataSet amb Kmeans (on K = 3)</i>	44
<i>Figura 13: Estructura d'arbre de clustering jeràrquic</i>	45
<i>Figura 14: Exemple de Clustering basat en centres.</i>	46
<i>Figura 15: Exemple de clustering basat en densitat</i>	47
<i>Figura 16: Dades en l'espai PCA (PLOT PCA) sobre iris dataset.</i>	49
<i>Figura 17: Disseny de l'aplicació</i>	50
<i>Figura 18: Implementació de la base de dades</i>	51
<i>Figura 19: Taula "testusuario"</i>	52
<i>Figura 20: Taula "Test"</i>	52
<i>Figura 21: Taula "grupopreguntas"</i>	52
<i>Figura 22: Taula "Preguntes"</i>	53
<i>Figura 23: Taula "Respostes"</i>	53
<i>Figura 24: Interfície de login</i>	65
<i>Figura 25: Pantalla principal sense res seleccionat</i>	65

<i>Figura 26: Estructura principal de la interfície part de learn .....</i>	<i>66</i>
<i>Figura 27: Pestanya Summary d'un algoritme. ....</i>	<i>67</i>
<i>Figura 28: Taula de dades (iris).....</i>	<i>68</i>
<i>Figura 29: Interfície principal de la part de Test. ....</i>	<i>68</i>
<i>Figura 30: Panell lateral de la part de test.....</i>	<i>69</i>
<i>Figura 31: Panell central de preguntes. ....</i>	<i>70</i>
<i>Figura 32: Panell central de resultats.....</i>	<i>70</i>
<i>Figura 33: Panell central amb els resultats amb plot. ....</i>	<i>71</i>

# Índex de Taules

<i>Taula 1 : Resum temps fases del projecte .....</i>	<i>26</i>
<i>Taula 2 : Costos recursos humans projecte .....</i>	<i>27</i>
<i>Taula 3: Costos nets del treballador.....</i>	<i>27</i>
<i>Taula 4: Costos del hardware per el desenvolupament del projecte.....</i>	<i>28</i>
<i>Taula 5: Costos de recursos software del projecte. ....</i>	<i>29</i>
<i>Taula 6: Altres despeses relacionades amb el projecte. ....</i>	<i>29</i>
<i>Taula 7: Variacions del pressupost inicial del projecte. ....</i>	<i>30</i>
<i>Taula 8: Resum del pressupost total del projecte.....</i>	<i>31</i>

# Índex de Codi

<i>Codi 1 : Implementació de la interfície de test .....</i>	<i>54</i>
<i>Codi 2: Implementació pel canvi d'interfície "learn-&gt;test" .....</i>	<i>54</i>
<i>Codi3: funcions reactives getLearn i getTest. ....</i>	<i>55</i>
<i>Codi 4: mòduls dels diferents algoritmes, inclosos en el server. ....</i>	<i>55</i>
<i>Codi 5 : Implementació de l' interfície de login .....</i>	<i>56</i>
<i>Codi 6: Comprovació de credencials a la base de dades .....</i>	<i>56</i>
<i>Codi 7: Implementació d'una columna amb un atribut de selecció.....</i>	<i>57</i>
<i>Codi 8: Llista desplegable amb selecció de la tasca a realitzar .....</i>	<i>58</i>
<i>Codi 9: Implementació d'un panell condicional amb la selecció d'algoritme .....</i>	<i>58</i>
<i>Codi 10: Implementació del Panell amb pestanyes. ....</i>	<i>59</i>
<i>Codi 11: Implementació d'una part de la funció de "renderPlot" (agafar paràmetres).....</i>	<i>59</i>
<i>Codi 12: Implementació d'interfície de la part de test.....</i>	<i>60</i>
<i>Codi 13: Implementació de la f.reactiva que retorna un test aleatori de la base dades.....</i>	<i>60</i>
<i>Codi 14: Implementació de la consulta a la base de dades per extreure els resultats dels tests d'un usuari. ....</i>	<i>61</i>
<i>Codi 15: Implementació d'una part del renderPlot.....</i>	<i>61</i>



# Resums

## Català

Començarem per explicar el concepte de *Machine Learning*, aquest és una disciplina a mig camí entre la informàtica i l'estadística, que té com a objectiu desenvolupar algoritmes i tècniques que permetin a les diferents màquines aprendre, és a dir, es tracta de crear un programa que generalitzi comportaments a partir de dades empíriques d'un procés.

Avui en dia en molts camps de la societat està present el *Machine Learning* com per exemple la medicina, el big data, el reconeixement de la veu...

Es per aquest motiu que molts alumnes tenen curiositat i volen saber més sobre els conceptes de *Machine Learning*, en algunes de les assignatures de la FIB.

Aquest TFG és una interfície web amigable que està dirigit precisament aquests alumnes que volen fer alguna d'aquestes assignatures, per ajudar-los en el seu aprenentatge d'aquesta matèria i que els serveixi com a guia.

L'objectiu principal és doncs, que els alumnes tinguin una eina per el suport del aprenentatge de l'assignatura, amb exemples de cadascú dels algoritmes i un apartat de test per poder avaluar els seus coneixements.

## English

First of all, *Machine Learning* is a science between informatics and statistics which main objective is that a machine learns with some algorithms

Today, In the actual society, the *Machine Learning* is present in different places, for example: medicine, Big Data and Voice recognition

Is for this reason that a lot of students are curious about this, and they want to know more about the concepts of *Machine Learning* in some subject of FIB.

This Final degree project is a friendly web interface that is created just for this kind of students, to helping them in the learning of this subject and to have a guide.

The most important objective is then, that the students have a tool for support their learning on subject, with examples of each algorithm and one section for test their knowledge about *Machine Learning*

## Castellano

En primer lugar explicaré el concepto de *Machine Learning* que es una disciplina entre la informática y la estadística, su principal objetivo es, desarrollar algoritmos y técnicas que permitan a las máquinas a aprender.

Hoy en día es una una ciencia que está presente en muchos lugares, como por ejemplo, en la medicina, el big data y el reconocimiento de voz entre otros.

Es por eso que algunos de los alumnos de la FIB, tienen curiosidad por aprender estas técnicas en algunas de las asignaturas que se ofrecen en la facultad.

Este TFG es una interfaz web amigable que está pensada precisamente para cubrir esa necesidad, para poder ayudarlos en su aprendizaje y que les sirva como guía.

Dicho esto, el objetivo principal es que los alumnos tengan una herramienta de soporte para el aprendizaje de la materia, con ejemplos de varios algoritmos y con un apartado de autoevaluación para demostrar sus conocimientos.

# PRIMERA PART: INTRODUCCIÓ I PLANIFICACIÓ

# 1 - Introducció

Aquest projecte respon a l'oferta proposada pel professor Luís Antonio Belanche, pertanyent al departament de ciències de la computació, exposada en el racó.

La raó del projecte venia donada perquè hi ha un augment d'alumnes que volen tenir coneixements de *Machine Learning*, és per això que hi ha la necessitat de tenir una eina de suport per l'assignatura/es d'aquesta temàtica.

A més el desenvolupament d'aquest TFG podrà servir d'ajut per altres universitats i/o facultats que tinguin matèries de temàtica similar, per altra banda, al tractar-se d'una aplicació Web, també podria servir de suport i aprenentatge de "becaris" en diferents feines on es necessiti el suport d'una eina com aquesta.

Per tant l'objectiu principal del projecte és aconseguir implementar una aplicació web, interactiva, usable i amigable per tal de que tota aquesta gent pugui tenir un suport per aprendre *Machine Learning*.

Aquesta aplicació Web integrarà principalment dos seccions:

1. Secció de learning : En aquesta part l'usuari disposarà de diferents algorismes que podran ser executats. Mitjançant diferents gràfics es podrà observar els resultats, així com un suport explicatiu i una taula on es mostraran les dades emprades per l'execució. Aquestes dades hauran sigut seleccionades per part de l'usuari en primera instància, a més, l'usuari podrà escollir diferents tasques a realitzar, just abans d'escollir l'algoritme, que estan dividides en :
  - a. Classificació
  - b. Regressió
  - c. Clustering
  - d. Extracció de Característiques

2. Secció de Test : En aquesta part, l'usuari podrà escollir entre diferents nivells de dificultat ( “easy” , “medium” , “hard” ) i un nombre de preguntes per tal d'avaluar-se amb un test de coneixements. L'usuari en qualsevol moment, podrà comprovar tots els seus resultats de manera textual o si vol , de manera gràfica.

## 1.1 - Estat de l'art

Passarem a explicar els conceptes necessaris o les eines que utilitzem per el nostre projecte.

### 1.1.1 - *Machine Learning*

*Machine Learning* és una disciplina a mig camí entre la informàtica i l'estadística, que té com a objectiu desenvolupar algoritmes i tècniques que permetin a les diferents màquines aprendre, és a dir, es tracta de crear un programa que generalitzi comportaments a partir de dades empíriques d'un procés<sup>[1]</sup>.

Aplicacions del *Machine Learning*:

- Motors de cerca
- diagnòstic mèdic
- detecció de fraus amb targetes de crèdit
- anàlisi del mercat de valors
- classificació de seqüències d'ADN
- reconeixement de la parla
- robòtica
- mineria de dades
- Big Data.

En el camp del *Machine Learning* existeixen diferents tipus d'algoritmes:

Supervisat, no supervisat, coneixement bàsic, coneixement estructurat, inductiu, deductiu, simbòlic, connexionista[2].

## 1.1.2- R

R és un llenguatge de programació Orientat a Objectes, interpretat i no compilat.

La sintaxis de R[3] és molt simple i intuïtiva. Es tracta d'un llenguatge que s'utilitza per anàlisis estadístic i gràfic.

R està basat en el llenguatge S i és de software lliure.[4]

Al tractar-se d'un llenguatge que s'utilitza per anàlisis estadístic s'utilitza molt en camps com: investigació biomèdica, la bioinformàtica i les matemàtiques financeres

Com va dir Ross Ihaka, un dels creadors de R:

*R: A language for data analysis and graphics*

## 1.1.3 – Rstudio

RStudio és un entorn de desenvolupament integrat(IDE) per a R. L'IDE es subdivideix en una consola, un editor de ressaltat de sintaxi que permet l'execució de codi directa, així com eines per la gestió de l'espai de treball [5].

## 1.1.4 - RShiny

Shiny és un framework per fer aplicacions webs interactives en R. La característica principal és la programació reactiva.[6]

Algunes de les avantatges que ofereix Shiny són les següents :

- Construcció d'aplicacions webs usables amb poques línies de codi, sense necessitat d'utilitzar "JavaScript"
- Las aplicacions en Shiny tenen el concepte similar a "Google Drive" ja que són "vives", això vol dir que és una aplicació que modifica la sortida per tots els canvis que realitza l'usuari d'entrada, sense tenir que actualitzar el navegador.
- Funcionen en qualsevol entorn que contingui "R"
- Conté el framework de desenvolupament de templates "Bootstrap"[\[7\]](#)
- Conté "widgets" pre dissenyats per tal de poder mostrar la sortida tant de "Plots (renderPot)", "Tables (renderTable)" o text ("renderPrint")

### 1.1.5 - Aplicacions actuals

En la actualitat, si t'hi pares a buscar, hi pots trobar diferents mètodes per aprendre *Machine Learning*.

Aquests mètodes, no tenen perquè tenir un suport online, per exemple:

- Articles Acadèmics ( [Genetic Algorithms and Machine Learning](#) )
- Llibres ( [Bishop, C.M , Pattern recognition and Machine Learning](#) )
- Publicacions i Presentacions ( [Machine Learning and Data Mining](#) )
- Classes Presencials en una facultat ( [FIB - APA](#) )

Però a part d'aquests també sabem que existeixen suports online, per intentar aprendre algunes de les tècniques de *Machine Learning*.

Per exemple, en la següent imatge trobem una aplicació on podem comprovar el funcionament de "K-MEANS", aquesta aplicació es troba a la *gallery* de RShiny[8]

Iris k-means clustering

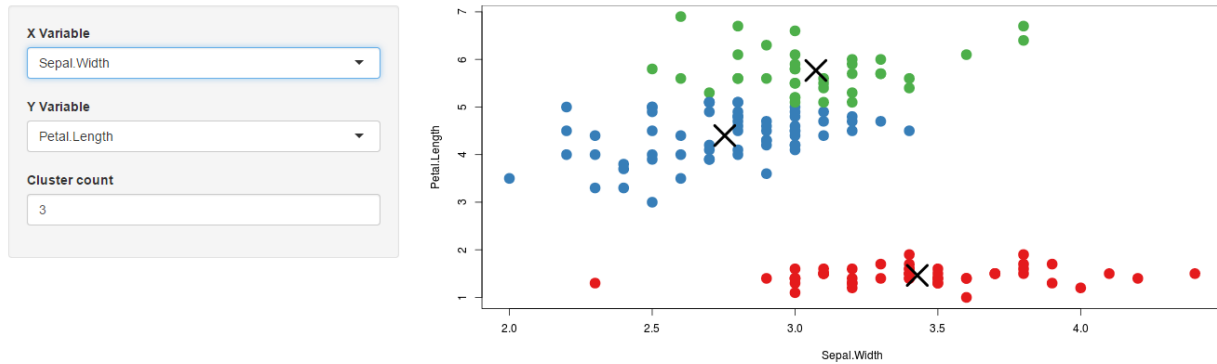


Figura 1 - Aplicació de learn K-means.

També tenim webs, on s'intenta donar suport per aprendre estadística, en la següent imatge es pot observar la web [IHateStatistics](#) .

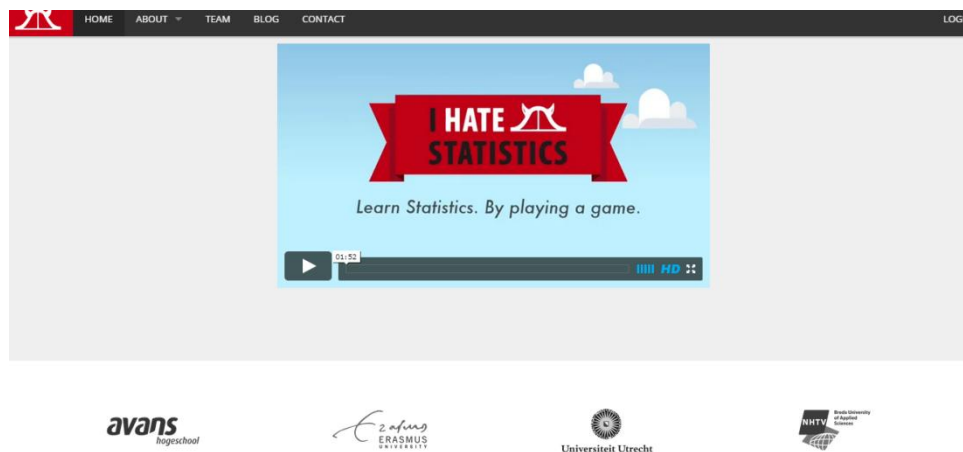


Figura 2: Web per aprendre estadística



Però si es vol una eina que contingui una visió més general i de més tipus d'algoritmes no hem trobat cap. Es per això que creiem que existeix la necessitat de desenvolupar aquesta eina, per poder cobrir la necessitat de tots els alumnes que volen aprendre d'aquesta temàtica.

## 2 - ABAST

### 2.1 - Objectius i requeriments

L'objectiu principal del projecte és crear una aplicació web que serveixi per aprendre *Machine Learning* a partir de poder practicar directament amb els algoritmes i de poder contestar preguntes i rebre respostes de manera independent del professor.

#### 2.1.1 - Objectius generals

A part tenim els següents objectius:


- Oferir una aplicació que sigui útil i usable.
- Facilitar el seguiment dels alumnes a l'assignatura.
- Que l'usuari pugui avaluar els seus coneixements.

#### 2.1.2 - Requeriments

Ja que per el desenvolupament del projecte, hem seguit la metodologia SCRUM, els requeriments del projecte, els definirem separant tots els stakeholders i mitjançant Històries d'usuari.

En les diferents HDU (Històries d'usuari) sinó s'indica el contrari suposarem que són requisits funcionals.

Una Història d'usuari, segueix la següent estructura:

Com a \_\_\_\_ (1) \_\_\_\_, vull que l'aplicació \_\_\_\_ (2) \_\_\_\_, per tal de \_\_\_\_ (3) \_\_\_\_ 

1. Usuari, persona que utilitza l'aplicació o administrador
2. Funcionalitat, requisits no funcionals .. .
3. Objectiu que es vol aconseguir.

#### 2.1.2.1 - Administrador / Professor :

- Com a administrador, vull que l'aplicació tingui dos seccions ben diferenciades de "learn" i de "test" per tal de poder oferir un correcte servei als usuaris.
- Com a administrador , vull que l'aplicació contingui diferents algoritmes de *Machine Learning* , per tal de que es puguin fer diferents proves.
- Com a administrador, vull que l'aplicació permeti tenir diferents usuaris i un control dels mateixos, per tal de que l'aplicació sigui utilitzable per tots ells.
- Com a administrador, vull que l'aplicació sigui escalable i fàcilment modificable , per tal de poder mantenir el sistema estable en "producció" (No funcional)

#### 2.1.2.2 - Usuari :

- Com a usuari , vull que l'aplicació em permeti entrar mitjançant nombre d'usuari i contrasenya, per tal de accedir a les funcionalitats de l'aplicació
- Com a usuari, vull que l'aplicació em permeti escollir diferents datasets, per tal de poder provar amb diferents dades els mateixos algoritmes.
- Com a usuari, vull que l'aplicació hem doni l'opció d'escollir entre diferents tasques (Classificació, regressió, Clustering o extracció de característiques) per tal de poder provar els diferents algoritmes.

- Com a usuari , vull que l'aplicació em permeti executar els diferents algoritmes que conté, per tal de veure el seu funcionament.
- Com a usuari, vull que l'aplicació em permeti intercanviar entre gràfics, text i taules en l'execució de cada algoritme, per tal de comprovar els valor de cadascú d'ells
- Com a usuari, vull que l'aplicació em permeti intercanviar entre les seccions de “learn” i “test” , per tal d'accedir-hi a ambdues.
- Com a usuari, vull que l'aplicació em permeti avaluar-me de diferents tests (“easy”, “medium”, “hard”) , per tal de conèixer el meu nivell en la matèria.
- Com a usuari, vull que l'aplicació em permeti comprovar tots els meus resultats de test, per tal de poder recuperar en qualsevol moment els resultats.
- Com a usuari, vull que l'aplicació em permeti veure un gràfic dels meus resultats, per tal de comprovar de manera més gràfica els meus resultats.
- Com a usuari, vull que l'aplicació respongui en un temps acceptable (menys de 5 segons) en l'execució dels algoritmes, per tal de no tenir que esperar massa temps davant cada pantalla. (No funcional)
- Com a usuari, vull que l'aplicació respongui eficientment els canvis de paràmetres que es facin, per tal de millora l'interacció amb l'aplicació (No funcional)

## 2.2 – Abast del TFG

L'aplicació és interactiva amb els usuaris i té tots els apartats necessaris, per tal de que l'alumne pugui seguir l'assignatura i que li serveixi de mètode de suport, per tal d'aprendre els diferents mètodes d'aquesta, i per tant de *Machine Learning*. Aquesta es divideix en dues parts, la primera és la part per aprendre, i una segona part és un mètode d'autoavaluació per comprovar els coneixements que hagi adquirit la persona.

La primera part té els diferents camps de *Machine Learning* (Regressió, Classificació, Extracció de característiques i Clustering), i la segona part és de caràcter general.

Aquesta aplicació no pretén ser un substitutiu de les classes presencials i no es cobert en l'abast d'aquesta, tampoc entra en l'abast que aquesta aplicació pugui ser utilitzada sense unes nocions prèvies.

## 3 - Metodologies

### 3.1 - Mètodes de treball

La metodologia que vam escollir per tal de desenvolupar aquest projecte, és a dir SCRUM<sup>[10]</sup>, ha sigut encertada, ja que en realitzar petites tasques i cicles curts de desenvolupament hem tingut una visió més realista del projecte i hem pogut detectar i resoldre problemes en un període més curt de temps. Hem tingut un feedback molt important per part del tutor del projecte, amb el qual hem realitzat quedades trisetmanals.

Els rols dins de la metodologia SCRUM van ser els següents:

1. "SCRUM MASTER" : En el nostre cas , el director del projecte, Luís Antonio Belanche.
2. "PRODUCT OWNER" : Podríem identificar com a product owner també al director del projecte, però també seria un stakeholder important el desenvolupador del projecte ja que hi té un interès en aquest.
3. "DEVELOPMENT TEAM" : En el nostre cas , aquest està format per mi, Sònia Sandalinas Pérez.

Com es pot observar, el director del projecte, comparteix dos càrrecs que segons la metodologia no seria correcte, però degut a les circumstàncies de que només hi ha dos membres hem hagut d'identificar-ho així.

## 3.2 - Eines de seguiment

Hem descartat l'opció d'un repositori GitHub ja que vam considerar que al haver-hi només un desenvolupador, és a dir, jo, no era necessari i implicaria massa costos. Per tant per controlar les diferents versions del projecte, hem utilitzat una estructura d'arbre dins del DropBox, separant cada versió per un nom diferent.

## 3.3 - Mètode de validació

Tenint en compte totes les proves i controls que s'han fet a cada versió, gràcies al sistema de control mitjançant DropBox i les reunions trisetmanals amb el tutor hem aconseguit la consecució de la realització del projecte amb tots els seus objectius.

Per comprovar la correcta usabilitat de l'aplicació, s'han fet diferents proves de l'aplicació a mesura que l'hem anat desenvolupant. Aquestes proves s'han portat a terme durant tota la fase de implementació del projecte.

## 3.4 - Algoritmes i programari

Per la realització d'aquest projecte sabem que les alternatives que teníem per escollir tenien que cobrir dos grans elements diferenciats.

En primer lloc, escollir un llenguatge de programació que tingués la potència necessària per tal d'executar algoritmes de *Machine Learning*, per altra banda, necessitàvem mostrar aquests resultats en una interfície Web amigable.

### 3.4.1 - Llenguatges de programació :

Per tal de poder executar els algoritmes de *Machine Learning*, amb un cost inferior en temps i complexitat es va fer un anàlisi d'alternatives de diferents llenguatges de programació.

En aquest anàlisi es van escollir entre quatre llenguatges diferents de programació : R , Java , MatLab i C++.

Dels seleccionats, els primers en ser descartats van ser Java i C++ , per la complexitat que comportava desenvolupar els diferents algoritmes de *Machine Learning* amb l'increment de temps que això suposava, a més , l'execució era més costosa.

Un cop teníem sobre la taula R i MatLab<sup>[11]</sup>, vam descartar aquest últim (MatLab) pel fet de que era un llenguatge propietari.

Per tant, finalment, vam escollir R com a llenguatge de programació ja que aquest esta pensat per computació estadística i per això els algoritmes de *Machine Learning* estan especialment ben programats.

### 3.4.2 - Aplicació Web :

Per la realització de la aplicació Web , és a dir, la interfície que visualitzaria l'usuari final i que hauria de ser amigable, teníem també quatre opcions diferents ( R (Shiny) , Python, PHP , Node.js)

Primerament vam descartar PHP i node.js pel fet de que la integració amb R era bastant complexa i no ens ofería tampoc gran potència per fer interfície gràfica, ja que són llenguatges més de “back-office”.

Estàvem entre Python i Shiny, un framework per fer aplicacions web amb R.

Després d’investigar una mica cadascuna de les alternatives, vam comprovar que Shiny ens ofería més facilitat a l’hora d’integrar , sobretot pel fet de comptar amb funciones reactives(1) i la seva organització en mòduls(2) a més de tenir integrat el framework de visualització CSS3 “bootstrap”.

## 4 - Planificació Temporal

Respecte a la planificació inicial, el calendari ha patit un notable endarreriment respecte el que es pretenia a primera instància. Les hores previstes en les diferents fases del projecte eren les encertades. Però no he pogut invertir tot el temps que havíem previst inicialment per portar el terme el projecte al llarg del curs.

Per aquest motiu, la planificació s’endarrereix una mica (per aquest motiu presentem el 25 de maig i no el abril com teníem previst), per tant les fases descrites a continuació han sofert un endarreriment en el temps , que no en les hores de treball assignades.

### 4.1 - Fases del projecte

#### 4.1.1 - Planificació i viabilitat del projecte.

Aquesta fase és la que es va portar a terme al inici del curs acadèmic, és a dir , el GEP. En aquesta fase es va dividir en els següents aspectes:

1. Abast
2. Planificació temporal
3. Gestió econòmica i sostenibilitat
4. Presentació preliminar
5. Contextualització i bibliografia
6. Plec de condicions
7. Documentació final

#### 4.1.2 - Repàs i estudi dels conceptes de *Machine Learning* (10/10/2014 - 19/12/2014)

En aquesta fase es van repassar els conceptes de *Machine Learning*, donats a l'assignatura d'APA(Aprenentatge Automàtic,és a dir *Machine Learning*), i que són imprescindibles per realitzar l'aplicació web. A més a més, es van estudiar els nous conceptes de *Machine Learning*, necessaris per l' aplicació, però no impartits en l'assignatura.

Amb aquests conceptes repassats i estudiats, teníem tot el necessari per desenvolupar la part tècnica del projecte.

#### 4.1.3 – Recull d'informació i estudi de les noves tecnologies (19/12/2014 - 2/01/2015)

En aquesta fase vam estudiar les eines per tal de desenvolupar el projecte (RShiny, RStudio) i vam recollir informació, amb exemples i manuals, per tenir els coneixements necessaris de les eines que necessitàvem.



#### 4.1.4 - Disseny de l'aplicació web (02/01/2015 - 22/01/2015)

En aquesta fase es va dissenyar com és la nostre aplicació. El disseny era tant gràfic, com de l'arquitectura del software.

#### 4.1.5 – Programació i testing de l'aplicació (22/01/2015 - 15/04/2015)

En aquesta fase vam desenvolupar l'aplicació, les dues parts: la part de learn i la part test. Ho vam fem amb els coneixements repassats en les fases anteriors.

Vam considerar tenir aquestes dues fases juntes, ja que el tractar-se d'un projecte amb petits objectius, vam fer la programació i el testing en paral·lel, ja que així trobaríem més ràpidament els errors i ens seria més fàcil corregir-los.

#### 4.1.6 - Part final del projecte (16/04/2015 - 25/05/2015)

En aquesta última etapa ja està finalitzat el software del projecte.

S'està creant un manual d'usuari i la memòria del projecte i es realitzarà la corresponent presentació davant del tribunal.

### 4.2 - Resum temps del projecte

Etapa	Hores
Planificació i viabilitat del projecte	75
Repàs i estudi dels conceptes de <i>Machine Learning</i>	150
Recull d'informació i estudi de les noves tecnologies	30
Disseny de l'aplicació web	30

Programació i testing de l'aplicació	150
Part final del projecte	60
TOTAL	495

Taula 1 : Resum temps fases del projecte

## 4.3 – Diagrama de Gantt Modificat

El diagrama de Gantt es troba en l'annex

# 5 - Gestió econòmica i pressupost

## 5.1 - Identificació dels costos

Podem resumir els costos involucrats en aquest projecte com a quatre: recursos humans, de hardware, software i generals.

Els costos que s'especifiquen a continuació són les previsions fetes per el desenvolupament del projecte, que han sofert petites variacions que s'expliquen al final d'aquest capítol.

## 5.2 - Costos recursos humans

Aquest projecte ha estat desenvolupat per una única persona, per tant s'ha encarregat de fer de "project manager", desenvolupador, desenvolupador "tester" i enginyer "software". Per tant diferenciarem de les 495 hores qui ha fet quines.

Persona	Hores estimades	Preu per hora Estimat (brut)	Total Estimat
Project manager	75	50 €	3750 €
Desenvolupador	315	35 €	11025 €
Desenvolupador Tester	75	30 €	2250 €
Enginyer software	30	40 €	1200 €
Total	495		18225 €

Taula 2 : Costos recursos humans projecte

Encara que els costos estan pensats per les hores de treball, i no per mesos, suposem que hi ha un únic pagament de 18.225€ en brut i que per tant es necessari fer-li les retencions corresponents.

Sou resultant del treballador	Preus
Sou brut total	18.225
-4,70% Cotització a la seguretat social	856,57
-1,55% Cotització a l'atur	282.48
-18% IRPF	3280.5
TOTAL	13805.45

Taula 3: Costos nets del treballador

## 5.3 – Costos de Hardware

Per desenvolupar aquest projecte necessitem un portàtil. A continuació mostrem el cost del hardware.

Producte	Preu	Unitats	Vida útil	Amortització estimada
Asus A550LD-XX375-H	699 €	1	4	87,37 €
Total	699 €			87,37 €

Taula 4: Costos del hardware per el desenvolupament del projecte

## 5.4 – Costos Software

Per poder realitzar el projecte necessitem un seguit de software. A continuació detallem el cost d'aquest.

Producte	Preu	Unitats	Vida útil	Amortització Estimada
Adobe Reader X	0€	1	-	0€
Windows 8.1	0€ (inclòs en portàtil)	1	-	0€
Microsoft Office 2013	139€	1	2	34.75€
Dropbox	0€	1	-	0€
Drive (Google)	0€	1	-	0€

RStudio	0€	1	-	0€
R	0€	1	-	0€
Shiny (R)	0€	1	-	0€
Total	139€	-	-	34.75€

Taula 5: Costos de recursos software del projecte.

## 5.5 – Altres despeses

A part de les despeses pròpies del projecte: sous, hardware i software, tenim altres despeses de caràcter general, que es detallen a continuació.

Concepte	Cost Estimat Mensual	Cost Estimat Total
Local	450 €	2700 €
Aigua	30 €	180 €
Llum	50 €	300 €
Internet	40 €	240 €
Total	570 €	3420 €

Taula 6: Altres despeses relacionades amb el projecte.

## 5.6 - Variacions del pressupost inicial

Les despeses del projecte no han variat respecte la planificació inicial ja que les hores han sigut les mateixes, per contra, les despeses generals com són Llum, aigua, local ... Si que han variat degut a l'ampliació d'un mes per l'entrega del projecte. El pressupost per tant és el següent :

Concepte	Cost Estimat Inicial	Cost Estimat Final
Local	2700€	3150€
Aigua	180€	210€
Llum	300€	350€
Internet	240€	280€
Total	3420€	3990€

Taula 7: Variacions del pressupost inicial del projecte.

## 5.7 – Pressupost total

Per tant , el pressupost final del projecte després de les variacions explicades en el punt anterior , queda de la següent forma :

Concepte	Cost
Recursos Humans	13805.45€
Hardware	87.37€
Software	34.75€
Despeses Generals	3990€
Total	17917.57€

Taula 8: Resum del pressupost total del projecte.

## 6 - Sostenibilitat i compromís social

### 6.1 Valoració de la sostenibilitat del projecte

La viabilitat econòmica d'aquest projecte és relativament alta, ja que els costos invertits en la realització del mateix han sigut mínims i la rendibilitat que se li pot treure pot ser molt alta, tant en un ambient acadèmic, com en un possible ambient laboral com s'ha comentat anteriorment.

A més, els costos de material son mínims, ja que és un projecte purament de programació, l'únic cost ha sigut l'adquisició d'un portàtil per poder desenvolupar l'aplicació i per tant, l'efecte ambiental no és elevat ja que aquesta inversió en el portàtil, continuarà sent rentable, dintre de 4 anys.

A més a més, un cop finalitzada la vida útil del material, aquest pot ser donat a diferents organitzacions per tal de que es reaprofiti i que hi hagi un menor cost ambiental.

Finalment, la viabilitat social d'aquest projecte és també bona, ja que amb la seva implementació es podrà millorar la vida del professorat, facilitant les classes, i per altra banda la de l'estudiant ja que podrà aprendre de forma més dinàmica i ràpida.

A més a més, al tenir un apartat de "test" l'aplicació podria servir com a mètode alternatiu d'avaluació repercutint en un percentatge de la nota.

En quant els aspectes econòmics , en aquest projecte no hem tingut cap col·laboració amb cap altre entitat, ni empresa ni acadèmica.

En el cas de que aquest projecte es desenvolupés en una empresa i no en l'àmbit acadèmic, considero que es podria portar a terme, seguint el pressupost establert.

El temps de realització de cada tasca és coherent segons la seva importància, i la planificació és prou ajustada com per poder realitzar el projecte en menor temps.

En quant els aspectes socials, com ja hem dit aquest projecte està dirigit a alumnes d'alguna assignatura de *Machine Learning*, no és un col·lectiu molt ampli de la societat. Però podria servir per realitzar millores en el diagnòstic mèdic, millores en detecció de targetes falses, i

milliores en molts altres camps en els quals s'utilitza el *Machine Learning*. Per tant l'impacte directe del projecte és baix, però considerant tots els àmbits en que s'aplica el *Machine Learning* l'impacte pot arribar a ser considerable.

## 6.2 - Matriu de sostenibilitat

¿Sostenible?	Econòmica	Social	Ambiental
Planificació	Viabilitat Econòmica	Millora en la qualitat de vida	Anàlisis de Recurs
Valoració	8	10	8
Resultat	Cost Final versus Previsió	Impacte en entorn Social	Consum de recursos
Valoració	8	10	9
Riscos	Adaptació a canvis d'escenari	Danys Socials	Danys Ambientals.
Valoració	-2	0	-1
Valoració Total	50		



# SEGONA PART: Descripció i Implementació

# 1 - Definició de conceptes principals

## 1.1 - Descripció del problema

El nostre problema consistia en el desenvolupament d'una aplicació web per aprendre *Machine Learning*.

Aquesta aplicació consta de diferents parts que definirem breument en aquest apartat.

### 1. Sistema de Login

En la nostra aplicació existeix un sistema de "login" d'usuaris consistent en una base de dades creada amb el sistema "MySQL" i controlada mitjançant el software "MySQLWorkbench", l'usuari tindrà un nom d'usuari, consistent en el seu correu electrònic i una password, aquestes dades seran generades pel administrador i subministrades als usuaris.

Un cop l'usuari inserta el nom i la contrasenya, el sistema valida que aquest usuari existeix a la Base de Dades i si les dades són correctes, li dóna accés a l'aplicació

#### Login

Account

Password

Figura 3 : Sistema de Login de l'aplicació

## 2. Part de "learn" :

Un cop l'usuari ha entrat amb els seus credencials correctament se li mostrarà la part de learn.

En aquesta part l'usuari disposarà de totes les funcionalitats per tal de tenir el suport per aprendre *Machine Learning*, en primera instància, l'usuari únicament veurà una llista desplegable, però a mesura que avanci en els passos, arribarà a una pantalla com es mostra en la següent figura.

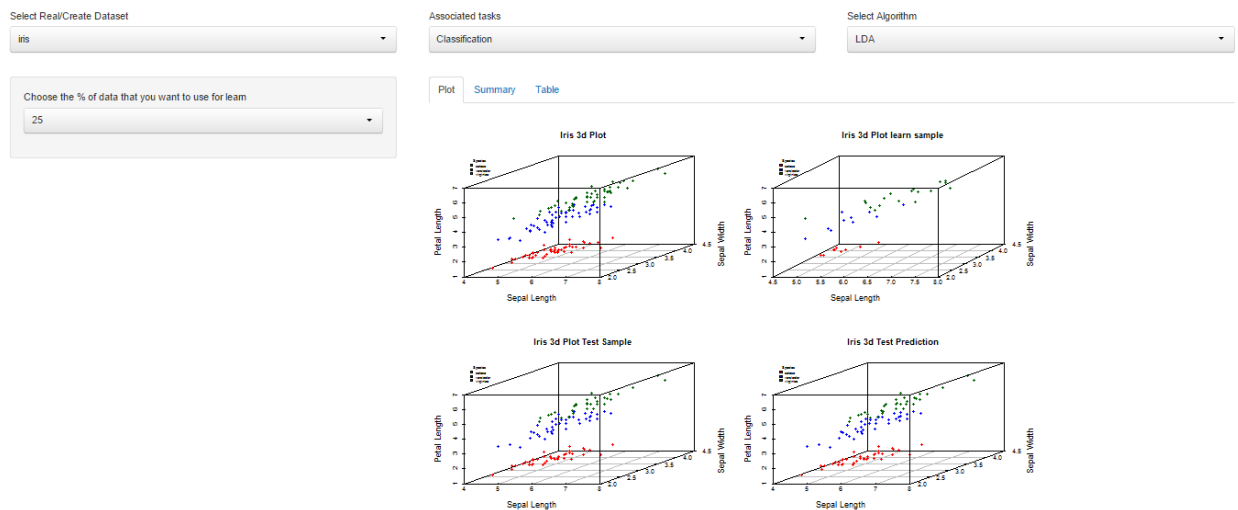


Figura 4 : Imatge sobre la part de "learn" de l'aplicació

## 3. Part de "test" :

En aquesta part, l'usuari es podrà examinar dels conceptes adquirits mitjançant les classes de l'assignatura així com també el suport donat per l'aplicació.

També podrà observar els seus resultats en format gràfic i de text.

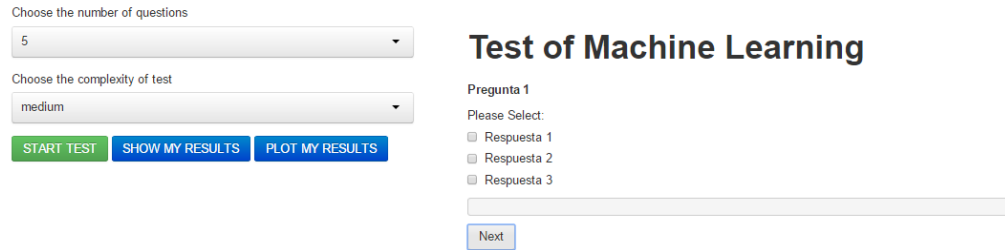


Figura 5 : Imatge sobre la part de “test” de l’aplicació

## 2 - Definició de les “Associated Tasks”

Hem classificat els algoritmes de *Machine Learning* en diferents tasques [\[12 .. 21\]](#) :

- Classificació
- Regressió
- Clustering
- Extracció de característiques

A continuació passarem a explicar cadascuna d’elles.

### 2.1 - Classificació

#### 2.1.1 - Conceptes

En *Machine Learning* anomenem classificació a la tasca que utilitzem per a cada element d’una col·lecció, assignar-lo a una categoria o classe, aquest és un problema generalitzat que engloba aplicacions diverses. En són un exemple la detecció de correus electrònics d’spam, la categorització de cèl·lules malignes o benigne basant-se en resultats de ressonàncies, com també la classificació de diverses galàxies, entre altres.

Aquesta és una tasca de aprenentatge supervisat(3).

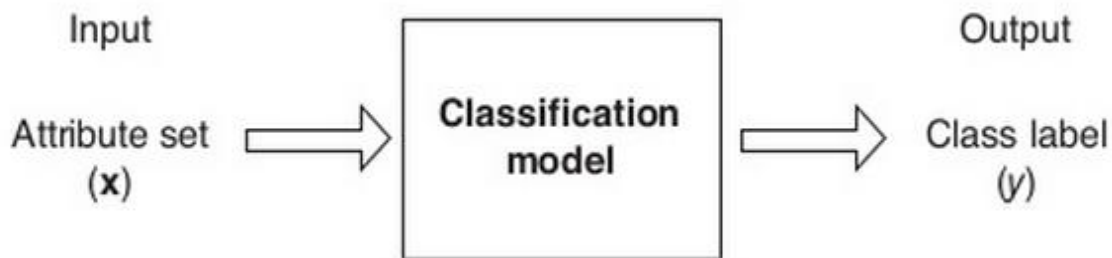


Figura 3: Definició gràfica de la tasca de Classificació.

L'objectiu de la tasca de classificació és predir amb exactitud la classe o categoria de cadascú dels elements de la col·lecció.

Per tant, les dades d'entrada per a una tasca de classificació és una col·lecció de registres, cada registre, es caracteritza per una tupla  $(x, y)$ , on  $x$  és el conjunt d'atributs i  $y$  és l'atribut de classe o atribut de categoria.

	Sepal.Length	Sepal.width	Petal.Length	Petal.width	Species
56	5.7	2.8	4.5	1.3	versicolor
22	5.1	3.7	1.5	0.4	setosa
136	7.7	3.0	6.1	2.3	virginica
102	5.8	2.7	5.1	1.9	virginica
63	6.0	2.2	4.0	1.0	versicolor
21	5.4	3.4	1.7	0.2	setosa
133	6.4	2.8	5.6	2.2	virginica
121	6.9	3.2	5.7	2.3	virginica
20	5.1	3.8	1.5	0.3	setosa
57	6.3	3.3	4.7	1.6	versicolor

Figura 4: Exemple de iris dataset.

En aquest exemple el conjunt d'atributs inclou: Sepal.Length, Sepal.Width, Petal.Length, Petal.Width i l'atribut de classe seria en aquest cas Species.

Els atributs presentats a la taula són numèrics excepte l'atribut de classe que aquest sempre és un atribut discret. Aquesta és una característica clau que distingeix classificació de regressió. (Una tasca que explicarem en el següent apartat).

En el nostre cas utilitzem un model de classificació per predir la etiqueta de classe de registres desconeguts, els models de classificació també tenen altre utilitat, servir com a eina explicativa per diferenciar entre objectes de diferents classes.

Un model de classificació podem dir què és com una caixa negra que assigna automàticament una classe quan hi té un conjunt d'atributs amb un atribut de classe desconegut.

Suposem que tenim les següents característiques d'una flor(iris) que tenim que classificar.

	Sepal.Length	Sepal.width	Petal.Length	Petal.width	Species
1	5.1	3.5	1.4	0.2	?

Figura 5: Registre a classificar.

Per determinar el registre de classe d'aquest ,utilitzem un model de classificació obtingut a partir de les dades de la figura 7.

## 2.1.2 - Enfocament general per la resolució d'un problema de classificació

Una tècnica de classificació és un enfocament sistemàtic per a la construcció de models de classificació d'un conjunt de dades d'entrada. Exemples: LDA (*Linear Discriminant Analysis*), QDA (*Quadratic Discriminant Analysis*), *Naive Bayes*, *Neural Networks*, *Support Vector Machines*, *Nearest Neighbor*, *Decision Trees* entre altres.

Cada tècnica utilitza un algoritme d'aprenentatge per a identificar un model que millor s'ajusti a la relació entre el conjunt d'atributs i l'atribut de classe de les dades d'entrada.

Un bon model de classificació s'ha d'ajustar a les dades d'entrada però també ha de predir correctament l'atribut de classe de registres que no hagi tingut en compte. S'ha de buscar el punt mig perquè no sobre ajusti les dades.

Per tant, el més important en un model de classificació és generar models amb una bona capacitat de generalització, és a dir, models que prediguin correctament el atribut de classe de mostres que desconeix.

Fases d'un algoritme de classificació:

1. Learn/Training Set: El conjunt d'entrada, on totes les mostres tenen un atribut de classe definit i que per tant coneixem. Aquest conjunt, és el que s'utilitza per crear el model de learn (model de classificació).
2. Test Set : El conjunt de dades que es vol obtenir l'atribut de classe, ja que aquest és desconegut.
3. Aplicació de model de classificació: A partir del model obtingut del conjunt d'entrada i l'algoritme de classificació podem predir els atributs de classe de tots els registres del conjunt de test.

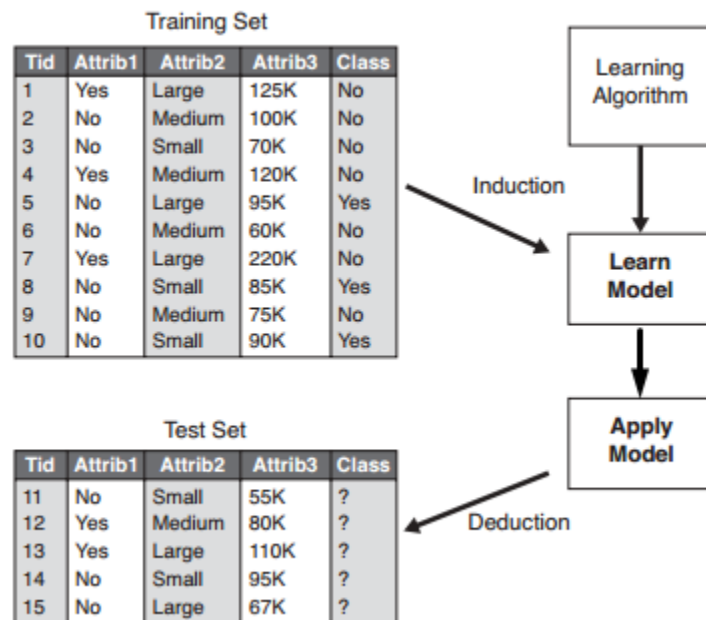


Figura 6: Resolució d'un problema de classificació.

Tan les dades del Learn Set com les dades del Test Set són extreptes d'un mateix dataset(4). En el cas de les dades de Test s'ha fet una extracció del seu atribut de classe, aquest atribut es guardat per després poder fer la comprovació de l'eficàcia del model ja que és l'atribut que es prediu.

### 2.1.3 - Eficàcia d'un model de classificació

Per avaluar l'eficàcia del nostre model tenim:

1. Matriu de resultats:

Consisteix en una matriu de la part de Test un cop ha estat predita, en aquesta es pot observar com ha estat predit el atribut de la classe per cadascuna de les diferents classes del model.

```
mydata.predict_test setosa versicolor virginica
setosa              12          0          0
versicolor          0          14          0
virginica            0          0          12
```

Figura 7: Matriu de resultats d'Iris Dataset.

Aquesta Matriu es pot interpretar de la següent manera: Les diagonals corresponen als valors que serien correctes del conjunt de dades original ja que equivalen a classe "x" i ha predit classe "x", les que estan fora de la diagonal, són valors incorrectes ja que ha predit un valor que no es el original.

En aquesta observació es pot comprovar que el model de classificació és molt bo, ja que totes les prediccions han sigut correctes, però en la majoria de casos, existeixen valors fora de les diagonals de la matriu ( que són un error ).

2. Eficàcia : Aquesta es calcula mitjançant la mitjana entre el nombre d'encerts , és a dir , el nombre de atributs de classe que ha predit correctament, dividit entre el nombre total de elements del conjunt de Test.

$$Eficàcia = \frac{\text{Encerts}}{\text{Total}}$$

## 2.2 - Regressió



## 2.2.1 - Conceptes

Donat un conjunt de dades, podem definir que aquest conjunt conté dos tipus de variables per cada registre:

- Variables Independents
- Variables Dependents

Aleshores, podem dir que en *Machine Learning*, regressió és la tasca d'obtenir una variable dependent a partir de la resta de variables del registre, la diferència amb Classificació es que la variable dependent és de tipus numèric.

L'objectiu de la tasca de regressió és predir el valor d'una variable dependent a partir dels valors de les altres variables del registre.

Per tant, les dades d'entrada per a una tasca de regressió són una col·lecció de registres, cada registre, es caracteritza per una tupla  $(x, y)$ , on  $x$  és el conjunt d'atributs i  $y$  és la variable dependent.

Aquesta és una tasca d'aprenentatge supervisat.

	x1	x2
44	0.88	0.726003817
7	0.14	0.004925153
74	0.98	1.066866777
103	0.56	0.166581594
138	1.26	1.386318207
43	0.86	0.437772675
21	0.42	0.380602954
126	1.02	0.976484887
69	0.88	1.076548379
29	0.58	0.158505952

Figura 8: Exemple de dataset on X2 depèn de X1.

En aquest exemple "X2" és una variable que depèn de "X1", per tant es la variable que volem predir amb el model de regressió.

Un model de regressió podem dir què és com una caixa negra que assigna automàticament un valor a una variable desconeguda.

Suposem que tenim les següents característiques.

	x1	x2
1	0.02	?

Figura 9: Registre a trobar el valor de la variable dependent "X2".

Per determinar el valor de "X2" d'aquest registre utilitzem un model de regressió obtingut a partir de les dades de la figura 10.

## 2.2.2 - Enfocament general per la resolució d'un problema de regressió

Com a exemples de tècniques de regressió tenim : Regressió lineal, Neural Networks, Suport Vector Machines, General linear model entre d'altres.

Cada tècnica utilitza un algoritme d'aprenentatge per a identificar un model que millor s'ajusti a la relació entre el conjunt de variables independents i la variable dependent.

Fases d'un algoritme de Regressió:

1. Learn/Training Set: El conjunt d'entrada, on totes les mostres són un conjunt de variables independents que coneixem. Aquest conjunt, és el que s'utilitza per crear el model de learn (model de regressió).
2. Test Set : El conjunt de dades que es vol obtenir el valor de la variable dependent, ja que aquest es desconegut.
3. Aplicació de model de regressió: A partir del model obtingut del conjunt d'entrada i l'algoritme de regressió podem predir els valors de la variable dependent de tots els registres del conjunt de test.

Tan les dades del “Learn Set” com les dades del “Test Set” són extrems d’una mateixa distribució.

En el cas de les dades de Test s’ha fet una extracció de la variable a predir, aquest valor es guardat per després poder fer la comprovació de l’eficàcia del model ja que és el valor que es prediu.

### 2.2.3 - Eficàcia d’un model de regressió

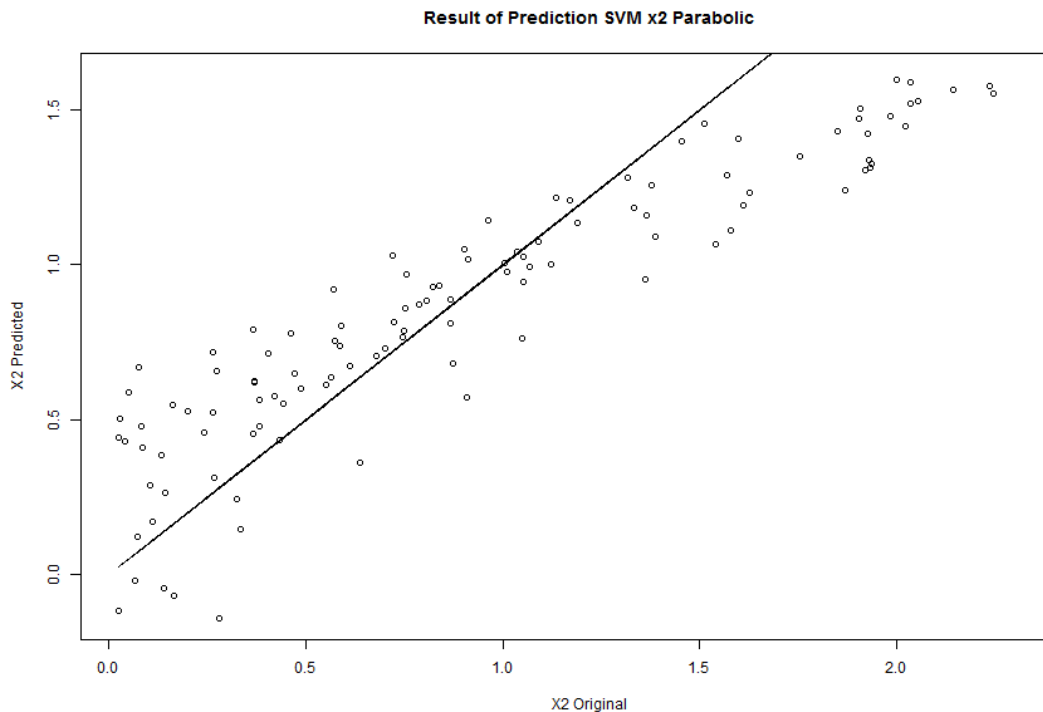


Figura 10: Resultat d’aplicar un model de regressió a una variable dependent “X2”

Per comprovar l’eficàcia d’un model de regressió s’utilitza l’error quadràtic, és a dir, a la mitja de les desviacions al quadrat de les prediccions dels valors reals.

## 2.3 - Clustering

### 2.3.1 - Conceptes

Podem definir el clustering com la assignació d’un conjunt d’observacions en subconjunts (clústers) de manera que les observacions en el mateix grup són similars.

El clustering és una tasca d'aprenentatge no supervisat (6).

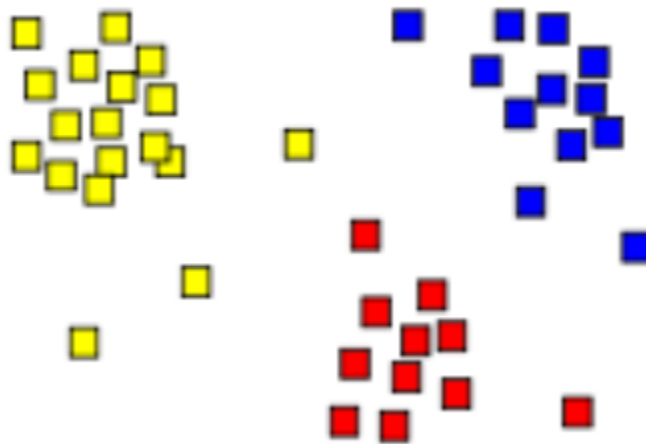


Figura 11: Exemple de clústers.

Els algorismes de clustering es basen en classificar o agrupar els registres en funció de les seves característiques en subgrups, per exemple "Kmeans" agrupa en K grups o clústers on K és un enter positiu.

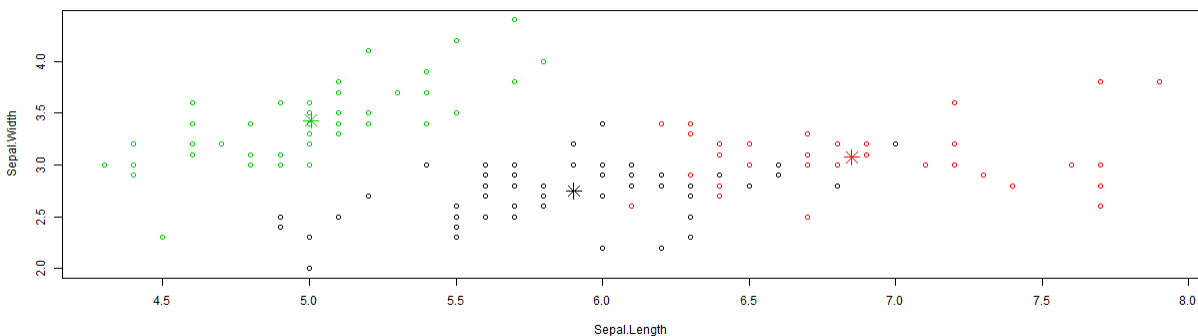


Figura 12: Exemple clústers en iris dataSet amb Kmeans (on K = 3)

El cost computacional de trobar la partició òptima d'N registres en K grups es NP-hard.

Podem agrupar el clustering en 4 tipus diferents:

- Clustering jeràrquic

En aquest mètode, les dades formen un diagrama d'arbre el qual es mostra la relació entre els objectes d'acord amb una matriu de proximitat. El node arrel del arbre representa tot el conjunt i cada node fulla és una única dada.

Els nodes intermitjos, descriuen com els objectes estan pròxims entre si.

Cada clustering és obtingut tallant l'arbre a diferents nivells. D'aquesta forma podem diferenciar dos formes de generar els clústers : tallant de l'arrel a les fulles ( divisiva) o inversament ( aglomerativa)

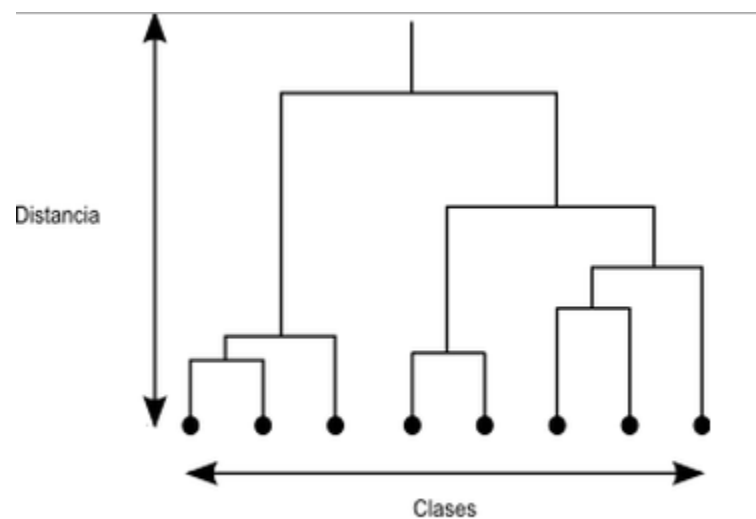


Figura 13: Estructura d'arbre de clustering jeràrquic

- Clustering basat en centres

Es tracta d'un algoritme iteratiu que particiona un conjunt de dades en un nombre de clústers especificats per el usuari. Comença assignant l'objecte representatiu de cada clúster (Centre) de forma aleatòria, agafat des de el conjunt de dades a agrupar.

A cada iteració, les dades són classificades assignant-les al clúster més proper.

Després de que totes les dades han sigut assignades, cada centre és recalculat.

El procés continua fins que cap clúster varií.

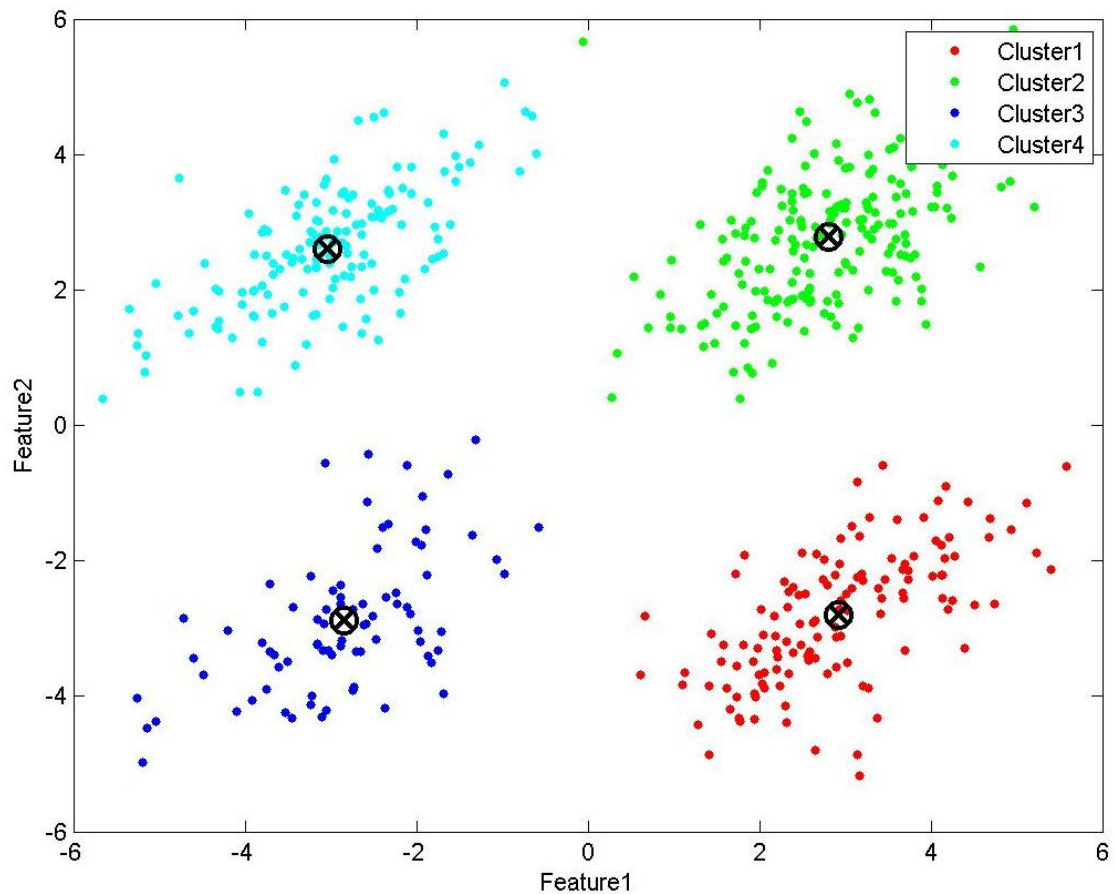


Figura 14: Exemple de Clustering basat en centres.

- Clustering basat en distribució

El Clustering basat en distribució és el model de agrupació més estretament relacionat amb l'estadística.

Els clústers es poden definir com una agrupació d'objectes que tenen una distribució semblant.

Una propietat convenient d'aquest tipus de clustering és que s'assembla a la forma en que es generen els conjunts de dades artificials: mitjançant el mostreig d'objectes al atzar d'una distribució

- Clustering basat en densitat

El clustering basat en densitat es podria definir com unes àrees on la densitat és més gran que en la resta del conjunt de dades.

Els objectes d'aquestes àrees disperses, que són requerits per separar els clústers, són normalment considerats com soroll i punts de frontera

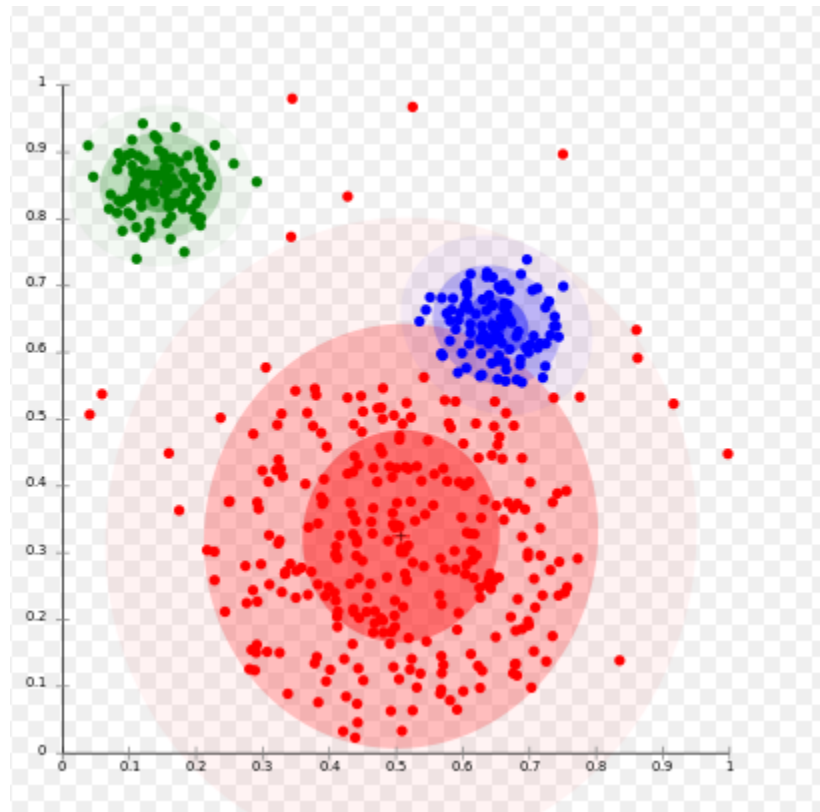


Figura 15: Exemple de clustering basat en densitat

### 2.3.3 - Eficàcia:

Com el objectiu del clustering és agrupar objectes similars en el mateix clúster i objectes diferents en diferents clústers, l'eficàcia es basa en aquests dos criteris:

- Cohesió: El membre de cada clúster ha de ser lo més pròxim possible a la resta de membres del clúster.

- Separació: Els clústers tenen que estar àmpliament separat entre ells.

## 2.4 - Extracció de Característiques

### 2.4.1 - Conceptes:

L'extracció de característiques és un mètode que pretén extreure la informació útil (característiques) i eliminar la informació redundant perquè després es pugui realitzar un procés (com per exemple de classificació).

L'extracció de característiques permetrà treballar sobre el problema de reconeixement de patrons utilitzant una quantitat de recursos computacionals menor que si haguéssim d'analitzar totes les dades. Tot i que això no ha de reduir la precisió del nostre mètode. Per a fer una bona extracció de característiques s'ha d'aconseguir eliminar la informació redundant o irrellevant, reduir la dimensionalitat del problema i maximitzar els aspectes més representatius de les dades perquè després es pugui realitzar una eficaç selecció de característiques i una posterior classificació.

Alguns exemples de mètodes d'extracció de característiques són : LDA (*Linear Discriminant Analysis* ), MCA( *Multiple Correspondence Analysis* ) , PCA ( *Principal Component Analysis* ) entre d'altres.



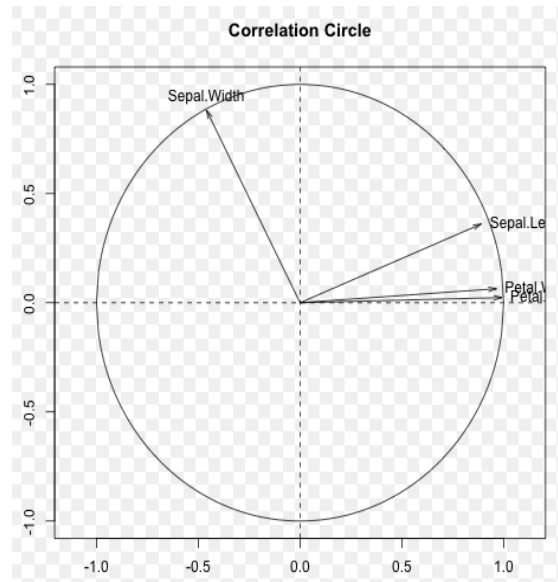


Figura 16: Dades en l'espai PCA (PLOT PCA) sobre iris dataset

## 3 - Disseny, Implementació I Satisfacció dels requisits

### 3.1 – Disseny

El sistema que hem utilitzat per a la nostre aplicació ha sigut una modificació del sistema en tres capes.

De la arquitectura en 3 capes, hem dividit en dos parts, la primera part que consisteix en la funcionalitat de vista i controlador la qual hem delegat al Shiny ja que té la potència necessària per tal de satisfer tots els nostres requeriments i la segona part, consistent en la generació d'un model de dades basat en un SGBDR (Sistema gestor de bases de dades relacionals) , que en el nostre cas ha estat MySQL.

En la següent imatge es pot comprovar com ha quedat l'estructura de la nostra aplicació que seguidament explicarem.

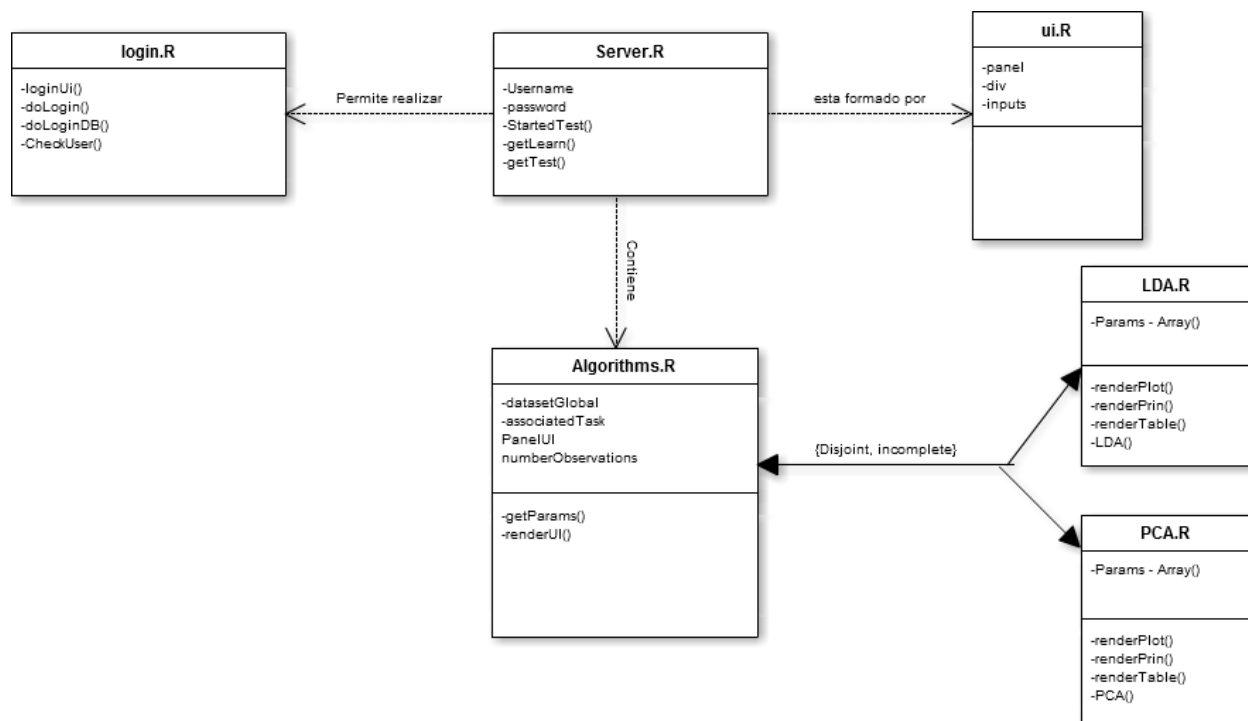


Figura 17: Disseny de l'aplicació

Com es pot observar i com hem comentat hem delegat la part de vista i controlador al Shiny, aquestes parts serien el “Server.R” i el “ui.R”.

Tenim un mòdul “Server.R” que s’encarrega de gestionar tota l’aplicació, ja que totes les interaccions han de passar per ell.

Per altra banda, tenim una interfície d’usuari principal que és la que generem a través del mòdul “ui.R” , com es pot observar, el “server.R” i el “ui.R” estan connectats, fent que d’aquesta forma qualsevol interacció de l’usuari provoqui el llançament d’un event al nostre Server per poder-ho controlar.

A part dels dos mòduls principals (“Server.R” i “ui.R”), tenim un mòdul que s’encarrega del sistema de “logging”, en aquest mòdul com es pot observar, tenim diferents funcions, algunes per crear una interfície i altres per la lògica del sistema amb la base de dades.

Per acabar, tenim diferents mòduls que els podem englobar en “Algorithms.R”, en aquests mòduls com en el cas del “logging” tenim també creacions d’interfícies i tota la lògica del algoritme en qüestió. D’aquest mòdul es generen tots els mòduls de cadascú dels algoritmes.

La segona part, consisteix en l’estructura de la base de dades, aquesta estructura ha estat creada i dissenyada tenint en compte totes les necessitats del projecte.

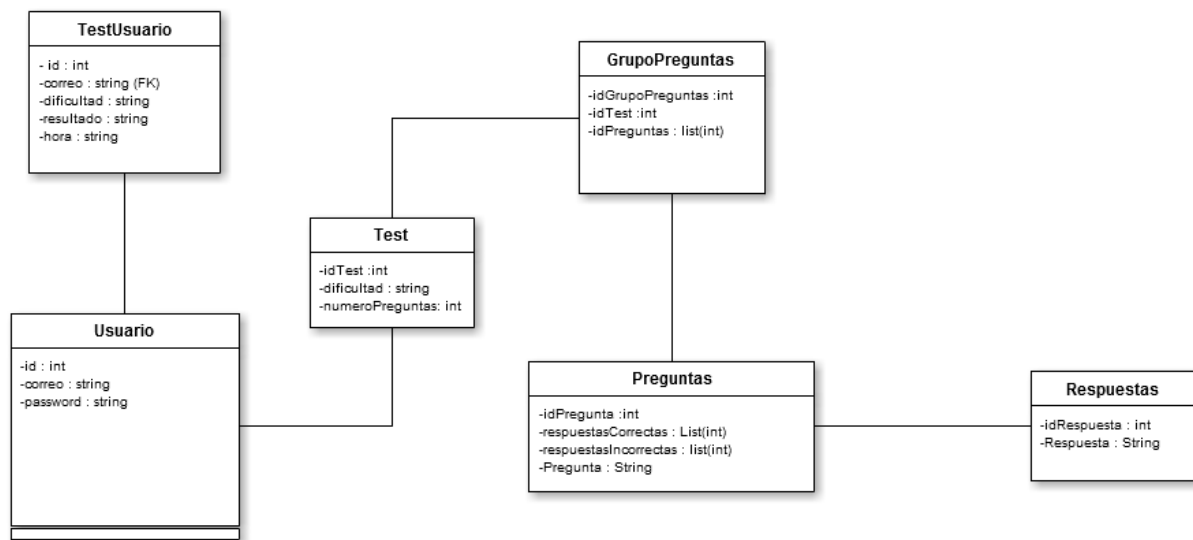


Figura 18: Implementació de la base de dades

La nostra base de dades esta formada per 6 taules, 5 dels quals són utilitzades per la realització de la part de test i l’altre per gestionar els usuaris.

A continuació expliquem cadascuna de les diferents taules.

En primer lloc la taula “testusuario” que és la taula on guardarem tota la informació referent als “tests” que l’usuari hagi anat realitzant a l’aplicació

Column	Type
◇ id	int(11)
◇ correo	varchar(250)
◇ dificultad	varchar(250)
◇ resultado	varchar(250)
◇ hora	varchar(200)

Figura 19: Taula "testusuario"

En aquesta taula principalment es faran insercions i consultes, cada cop que un usuari finalitzi un test, es guardarà el seu resultat amb la hora a la base de dades , per, posteriorment, sigui consultat quan es premi el botó de "show my results" o "Plot My Results"

La resta de taules són totes necessàries per tal de fer un test modular, que pugui variar de nombre de preguntes i respostes.

En primer lloc la taula "test" on guardarem tots els tests que han estat creats amb un identificador

◇ idTest	int(11)
◇ dificultad	varchar(45)
◇ numeroPreguntas	int(11)

Figura 20: Taula "Test"

En segon lloc, un cop tenim un test definit, per aquest "test" definirem un grup de preguntes.

◇ idgrupoPreguntas	int(11)
◇ idTest	varchar(45)
◇ idPreguntas	varchar(250)

Figura 21: Taula "grupopreguntas"

Com es pot observar en la imatge anterior, un grup de preguntes esta format per una cadena de preguntes.

◇ idPregunta	int(11)
◇ respuestasCorrectas	varchar(220)
◇ respuestasIncorrectas	varchar(220)
◇ pregunta	varchar(500)

Figura 22: Taula "Preguntes"

I cada pregunta està formada per una cadena de respostes correctes, respostes incorrectes i la pregunta en format text.

I per acabar, la taula de respostes, que està formada per la resposta en format text

◇ idRespuesta	int(11)
◇ respuesta	varchar(500)

Figura 23: Taula "Respostes"

## 3.2 – Implementació i Satisfacció de Requisits

Com hem comentat a la primera part, tenim una sèrie de requisits que el nostre projecte tenia que satisfer, per tant els anirem explicant un per un i també indicarem com encaixa la resolució dins el nostre sistema.

### 3.2.1 - Administrador / Professor :

- 1.1 - Com a administrador, vull que l'aplicació tingui dos seccions ben diferenciades de "learn" i de "test" per tal de poder oferir un correcte servei als usuaris.
  - Aquest requisit s'ha complert gràcies a la implementació de dos interfícies separades, una per la part de "test" i una per la part de "learn" ambdues

en dos mòduls separats, a més, s'ha integrat en el server una funció reactiva que al clicar un botó es canvia entre aquestes interfícies.

```
testUI <- function (){  
  div(class= "",  
    selectInput("questions","Choose the number of questions", c("5","10", "15")),  
    selectInput("complexity","Choose the complexity of test",c('easy','medium','hard')),  
    div(class="",  
      shiny::actionButton("start_test", "START TEST", styleclass = 'success'),  
      shiny::actionButton("show_history","SHOW MY RESULTS", styleclass='primary'),  
      shiny::actionButton("show_plot","PLOT MY RESULTS", styleclass='primary')  
    )  
  )  
}
```

Codi 1 : Implementació de la interfície de test

Com es pot observar la nomenclatura es semblant a html5 i bootstrap, hem utilitzat “acion\_buttons” per les diferents funcions.

```
output$mainPage <- renderUI({  
  doLoginBD()  
  doLogout()  
  getTest()  
  getLearn()
```

Codi 2: Implementació pel canvi d'interfície “learn->test”

Com s'observa a la imatge anterior en el nostre mòdul principal (“server”) tenim les crides reactives per detectar la pulsació de l'usuari a qualsevol dels botons de “learn” o “test”

```

getTest <- reactive({
  if (!is.null(input$test) && input$test > 0) {
    isolate(
      loginData$TestMode <- TRUE
    )
  }
})

getLearn <- reactive({
  if(!is.null(input$learn) && input$learn > 0){
    isolate(
      loginData$TestMode <- FALSE
    )
  }
})

```

Codi3: funcions reactives getLearn i getTest.

Per últim, amb aquestes comprovacions i mitjançant una variable global compartida amb el server podem canviar entre el “learn” i el “test”

- 1.2 - Com a administrador , vull que l’aplicació contingui diferents algoritmes de *Machine Learning* , per tal de que es puguin fer diferents proves.
  - Aquest requeriment s’ha complert mitjançant l’ implementació dels diferents algoritmes en l’aplicació.

```

source("PCA.R", local=T)
source("LDA.R", local=T)
source("QDA.R", local=T)
source("MCA.R", local=T)
source("NB.R", local=T)
source("MLP.R", local=T)
source("SVM.R", local=T)
source("KMEANS.R", local=T)

```

Codi 4: mòduls dels diferents algoritmes, inclosos en el server.

Com es pot comprovar en el nostre mòdul principal hi incloem els diferents mòduls per cada un dels algoritmes.

- 1.3 - Com a administrador, vull que l'aplicació permeti tenir diferents usuaris i un control dels mateixos, per tal de que l'aplicació sigui utilitzable per tots ells.
  - Aquest requisit s'ha complert mitjançant un sistema de "logging" en l'aplicació per mitja d'un formulari que s'integra en el mòdul de "login.R" i una funció que comprova els credencials de l'usuari a la base de dades.

```
# Login user interface ----
loginUI <- function (){
  div(class= "",
      code,
      textInput("account", "Account",value="example@example.com"),
      passwordTextInput("pwd",label = "Password"),
      shinyjs::ActionButton("login", "Login", styleclass='success')
  )
}
```

Codi 5 : Implementació de l' interfície de login

```
#login base de datos
doLoginBD <- reactive ({
  if (!is.null(input$login)) {
    if (input$login > 0) {
      con = dbconnect(MySQL(), user='root', password='1234', db='TFG', host='localhost')
      prueba <- input$account
      query = paste("select * from usuarios where correo =",prueba,"",sep='')
      result2<-dbGetQuery(con,query)
      dbDisconnect(con)
      pass <- input$pwd
      if(identical(pass,result2$password)){
        loginData$Account <- result2$correo
        loginData$LoggedIn <- TRUE
      }
    }
  }
})
```

Codi 6: Comprovació de credencials a la base de dades

- 1.4 - Com a administrador, vull que l'aplicació sigui escalable i fàcilment modificable , per tal de poder mantenir el sistema estable en "producció" (No funcional)
  - Aquest requisit s'ha complert gràcies a que l' implementació s'ha realitzar en mòduls i per tant són fàcilment modificables, a més com es un sistema web es pot escalar sempre que es vulgui simplement de forma horitzontal



afegint una nova instància i automàticament es duplicaria la quantitat d'usuaris acceptats.

### 3.2.2 - Usuari:

- 2.1 - Com a usuari, vull que l'aplicació em permeti entrar mitjançant nombre d'usuari i contrasenya, per tal de accedir a les funcionalitats de l'aplicació
  - Aquest requisit, és compartit amb l'administrador, es compleix mitjançant l'esmentat anteriorment.
- 2.2 - Com a usuari, vull que l'aplicació em permeti escollir diferents datasets, per tal de poder provar amb diferents dades els mateixos algoritmes.
  - Aquest requisit s'ha complert gràcies a l'implementació d'un sistema de atributs seleccionables.

```
column(4,br(),selectInput  
  (inputId = "datasetSelection",  
   label = "Select Real/Create Dataset",  
   choices = c("Select Dataset", "iris", "decathlon", "parabolic", "fibonacci", "tea", "hobbies"),  
   selected = selectedDataset()  
  ), ..  
), ..
```

Codi 7: Implementació d'una columna amb un atribut de selecció.

- 2.3 - Com a usuari, vull que l'aplicació em doni l'opció d'escollir entre diferents tasques (Classificació, regressió, Clustering o extracció de característiques) per tal de poder provar els diferents algoritmes.
  - Aquest requisit s'ha complert mitjançant la implementació d'un atribut de selecció (llista desplegable)

```
selectInput
(inputId = "typeSelection",
label = "Associated tasks",
choices = c("Select type", "Classification", "Regression", "Feature Extraction", "Clustering"),
selected = selectedType()
)
```

Codi 8: Llista desplegable amb selecció de la tasca a realitzar

- 2.4 - Com a usuari , vull que l'aplicació em permeti executar els diferents algoritmes que conté, per tal de veure el seu funcionament.
  - Aquest requisit s'ha complert mitjançant, en primer lloc, una llista desplegable ( atribut de selecció ) on es pot escollir el tipus d'algoritme segons quina tasca s'hagi escollit amb anterioritat, això s'ha aconseguit amb un "condicionalPanel"

```
conditionalPanel(
condition = "input.datasetSelection != 'Select Dataset'
&& input.typeSelection != 'Select Type'
&& input.datasetSelection != 'tea'
&& input.datasetSelection != 'hobbies'
&& input.typeSelection == 'Classification'",
selectInput
(inputId = "algorithm2",
label = "Select Algorithm",
choices = c("Select", "LDA", "QDA", "NBAYES", "MLP", "SVM"),
selected = selectedAlgorithm2()
)
),
```

Codi 9: Implementació d'un panell condicional amb la selecció d'algoritme

Un cop l'usuari escull l'algoritme en qüestió, s'executarà el mòdul corresponent creant d'aquesta forma l'interfície adequada.

- 2.5 - Com a usuari, vull que l'aplicació em permeti intercanviar entre gràfics, text i taules en l'execució de cada algoritme, per tal de comprovar els valor de cadascú d'ells
  - Aquest requisit s'ha complert gràcies a l' utilització de un "TabSetPanel" en el qual podem crear diferents pestanyes dintre d'un panell, per tant crearem una pestanya per cada funcionalitat, com Shiny funciona de forma reactiva, quan l'usuari intercanvia entre pestanyes el codi corresponent serà executat.

```
mainPanel(
  tabsetPanel(type = "tabs",
    tabPanel("Plot", plotoutput("plot_kmeans")),
    tabPanel("Plot original DataSet", plotoutput("summary_kmeans")),
    tabPanel("Table", tableoutput("table_kmeans"))
  )
)
```

Codi 10: Implementació del Panell amb pestanyes.

```
output$plot_mlp <- renderPlot({
  datasetSelected <- input$datasetselection
  typeAlgorithm <- input$typeselection # Clasificación o Regresión / Clasification/ Regresion
  porcentaje_obs <- as.numeric(input$observaciones)
  decay <- as.numeric(input$decay)
  size <- as.numeric(input$size)
  maxit <- as.numeric(input$maxit)
```

Codi 11: Implementació d'una part de la funció de "renderPlot" (agafar paràmetres)

- 2.6 - Com a usuari, vull que l'aplicació em permeti intercanviar entre les seccions de "learn" i "test" , per tal d'accedir-hi a ambdues.
  - Aquest requisit, és compartit amb l'administrador, es compleix mitjançant l'esmenta't en el requisit 1.1.
- 2.7 - Com a usuari, vull que l'aplicació em permeti avaluar-me de diferents tests ("easy", "medium", "hard") , per tal de conèixer el meu nivell en la matèria.

- Aquest requisit s'ha complert mitjançant una implementació que interactua entre el mòdul de server i la base de dades. En primer lloc en el server agafem els paràmetres que l'usuari ha escollit, un cop l'usuari prem "START TEST" una funció reactiva es cridada des de el server i s'encarrega de recuperar un test aleatori de la base de dades i muntar-lo en la interfície. Per finalitzar es guarden totes les respostes de forma global i quan l'usuari finalitza el test un altre funció reactiva que calcula el resultat es executada.

```
testUI <- function (){
  div(class= "",
    selectInput("questions","Choose the number of questions", c("5","10", "15")),
    selectInput("complexity","Choose the complexity of test",c('easy','medium','hard')),
    .....
```

#### Codi 12: Implementació d'interfície de la part de test

```
startedTest<- reactive({
  if(!is.null(input$start_test) && input$start_test > 0){
    con = dbConnect(MySQL(), user='root', password='1234', db='TFG', host='localhost')
    numPreguntas <- as.numeric(input$questions)
    dificultad <- input$complexity
    #Se escoge un Test Aleatorio segun los parametros de los usuarios
    query1 = paste("select idTest from test where dificultad =",dificultad, "",sep='')
    query2 = paste("and numeroPreguntas=",numPreguntas,"",sep='')
    query = paste(query1,query2,sep='')
    result2 <- dbGetQuery(con,query)
    ids <- result2$idTest
    .....
```

#### Codi 13: Implementació de la f.reactiva que retorna un test aleatori de la base dades.

- 2.8 - Com a usuari, vull que l'aplicació em permeti comprovar tots els meus resultats de test, per tal de poder recuperar en qualsevol moment els resultats.
  - Aquest requisit s'ha complert mitjançant una funció reactiva que es llançada quan es prem el botó de "SHOW MY RESULTS" , la qual busca a la base de dades tots els resultats del usuari i els mostra en una interfície.

```

output$myresults <- renderPrint({
  if(!is.null(input$show_history) && input$show_history > 0){
    con = dbConnect(MySQL(), user='root', password='1234', db='TFG', host='localhost')
    query = paste("SELECT dificultad, resultado,hora from testusuario where correo='", input$account,"'",sep='')
    results <- dbGetQuery(con,query)
    dbDisconnect(con)
    print(results)
  }
});

```

Codi 14: Implementació de la consulta a la base de dades per extreure els resultats dels tests d'un usuari.

- 2.9 - Com a usuari, vull que l'aplicació em permeti veure un gràfic dels meus resultats, per tal de comprovar de manera més gràfica els meus resultats.
  - Aquests requisit s'ha complert com el requisit 2.8 , la diferencia es que els resultats extrets de la base de dades i posteriorment es fa un plot mitjançant "barplot"

```

tot <- colMeans(p)
text(mp, tot + 3, format(tot), xpd = TRUE, col = "blue")
barplot(p$resultados_numeric, xlab = "Results of each test" , ylab = "score" ,
texto <- paste("Results of Test " , dificultad, sep="")
title(main = texto, font.main = 4)

```

Codi 15: Implementació d'una part del renderPlot

- 2.10 - Com a usuari, vull que l'aplicació respongui en un temps acceptable (menys de 5 segons) en l'execució dels algorismes, per tal de no tenir que esperar massa temps davant cada pantalla. (No funcional)
  - Aquest requisit s'ha complert gràcies a que R té una millor execució dels algorismes de *Machine Learning*, a més s'han fet comprovacions i cap algoritme supera els 4s de temps d'execució, tenint en compte el la part més costosa, en el nostre cas, l' impressió dels gràfics de l'intefície .

- 2.11 - Com a usuari, vull que l'aplicació respongui eficientment els canvis de paràmetres que es facin, per tal de millora l' interacció amb l'aplicació (No funcional)
  - Aquest requisit s'ha complert gràcies a l'habilitat de RShiny i les seves funcions reactives, ja que d'aquesta forma qualsevol canvi de l'usuari es monitoritzat en temps real.

## 4 - Assoliment competències Tècniques

Per el desenvolupament del nostre projecte teníem una sèrie de competències tècniques per assolir. Aquestes estan definides a continuació amb les raons per les quals creiem que han estat assolides.

1. CCO1.3: Definir, avaluar i seleccionar plataformes de desenvolupament i producció hardware i software per al desenvolupament d'aplicacions i serveis informàtics de diversa complexitat.
  - a. Nivell d'assoliment requerit : Una mica
  - b. Nivell d'assoliment adquirit : Una mica
  - c. Justificació: A partir de realitzar l'estudi de les diferents tecnològiques, aquest estudi s'ha realitzat mitjançant comparatives ("benchmarks") tant de llenguatges de programació com d' entorns de desenvolupament (IDE) , a més s'ha comprovat amb èxit que el software pot córrer amb un hardware no massa gran.
2. CCO2.1 : Demostrar coneixement dels fonaments, dels paradigmes i de les tècniques pròpies dels sistemes intel·ligents, i analitzar, dissenyar i construir sistemes, serveis i aplicacions informàtiques que utilitzin aquestes tècniques en qualsevol àmbit d'aplicació
  - a. Nivell d'assoliment requerit : Bastant

- b. Nivell d'assoliment adquirit : Bastant
  - c. Justificació : S'ha assolit mitjançant l'estudi de les diferents tècniques de *Machine Learning*, aquestes són necessàries per el correcte funcionament de l'aplicació, per tant com l'aplicació funciona correctament (*referència a Exemple d'execució de l'aplicació*), podem dir que hem assolit la competència.
3. CCO2.3 : Desenvolupar i avaluar sistemes interactius i de presentació d'informació complexa, i la seva aplicació a la resolució de problemes de disseny d'interacció persona computador
- a. Nivell d'assoliment requerit : Bastant
  - b. Nivell d'assoliment adquirit : Bastant
  - c. Justificació : S'ha assolit amb el desenvolupament de l'aplicació, ja que aquesta és interactiva amb l'usuari gràcies a l'implementació amb Shiny. També amb el fet de mostrar a l'usuari una informació útil i fàcil de comprendre.
4. CCO2.4 : Demostrar coneixement i desenvolupar tècniques d'aprenentatge computacional; dissenyar i implementar aplicacions i sistemes que les utilitzin, incloent les que es dediquen a l'extracció automàtica d'informació i coneixement a partir de grans volums de dades.
- a. Nivell d'assoliment requerit : Profund
  - b. Nivell d'assoliment adquirit : Profund
  - c. Justificació : S'ha assolit gràcies al desenvolupament de totes les tècniques de *Machine Learning* per tractar grans volums de dades, com són per exemple "K-means Clustering" , "NEURAL NETWORKS" i "SUPPORT VECTOR MACHNES", aquestes tècniques es poden comprovar amb l'execució de l'aplicació.
5. CCO2.6 : Dissenyar i implementar aplicacions gràfiques, de realitat virtual, de realitat augmentada i videojocs.

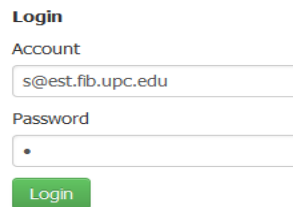
- a. Nivell d'assoliment requerit: Una mica
  - b. Nivell d'assoliment adquirit : Una mica
  - c. Justificació: Aquesta competència s'ha assolit mitjançant l' implementació de plots 3d en l'aplicació, juntament amb tot el desenvolupament de la part gràfica.
6. CCO3.1 : Implementar codi crític seguint criteris de temps d'execució, eficiència i seguretat.
- a. Nivell d'assoliment requerit : Bastant
  - b. Nivell d'assoliment adquirit : Bastant
  - c. Justificació : S'ha adquirit mitjançant la correcta implementació, tenint en compte aspectes de modularitat però sense deixar de banda els temps d'execució superiors que podria produir massa d'aquesta, s'han emprat bones pràctiques a l'hora del desenvolupament.
7. CCO3.2 : Programar considerant l'arquitectura hardware, tant en assemblador com en alt nivell.
- a. Nivell d'assoliment requerit : Una mica
  - b. Nivell d'assoliment adquirit : Una mica
  - c. Justificació : S'ha assolit gràcies a la prova de l'aplicació a diferents ordinadors, a més aquesta aplicació serà pujada al portal scienceNodes i per tant haurà sigut provada en un entorn de producció real.

## 5 - Exemple d'execució d'aplicació



## 5.1 – Exemple d’execució de la part de “Login”

El primer que ens trobem al obrir l’aplicació és amb la interfície de “Login”, en aquesta es pot observar com hi ha els camps necessaris i l’usuari ha d’insertar les seves credencials i fer “Login”



The screenshot shows a login form with the following elements:

- Login** (Section Header)
- Account** (Label) with an input field containing "s@est.fib.upc.edu"
- Password** (Label) with an input field containing a single dot "."
- Login** (Green button)

Figura 24: Interfície de login

Un cop introduïts per l’usuari aquests dos camps, el sistema llançarà la “query” contra la base de dades per comprovar si l’usuari introduït existeix i si les seves credencials són correctes.

Si és així el sistema li donarà accés a la pantalla principal de l’aplicació.

## **MACHINE LEARNING**



The screenshot shows the main application interface with the following elements:

- MACHINE LEARNING** (Section Header)
- User: s** (Text)
- Test Mode** (Blue button)
- Logout** (Blue button)
- Select Real/Create Dataset** (Text)
- Select Dataset** (Dropdown menu)

Figura 25: Pantalla principal sense res seleccionat

En aquesta imatge es pot observar l’ interfície que l’usuari veurà un cop autoritzat en l’aplicació

## 5.2 - Exemple d’execució de la part de learn

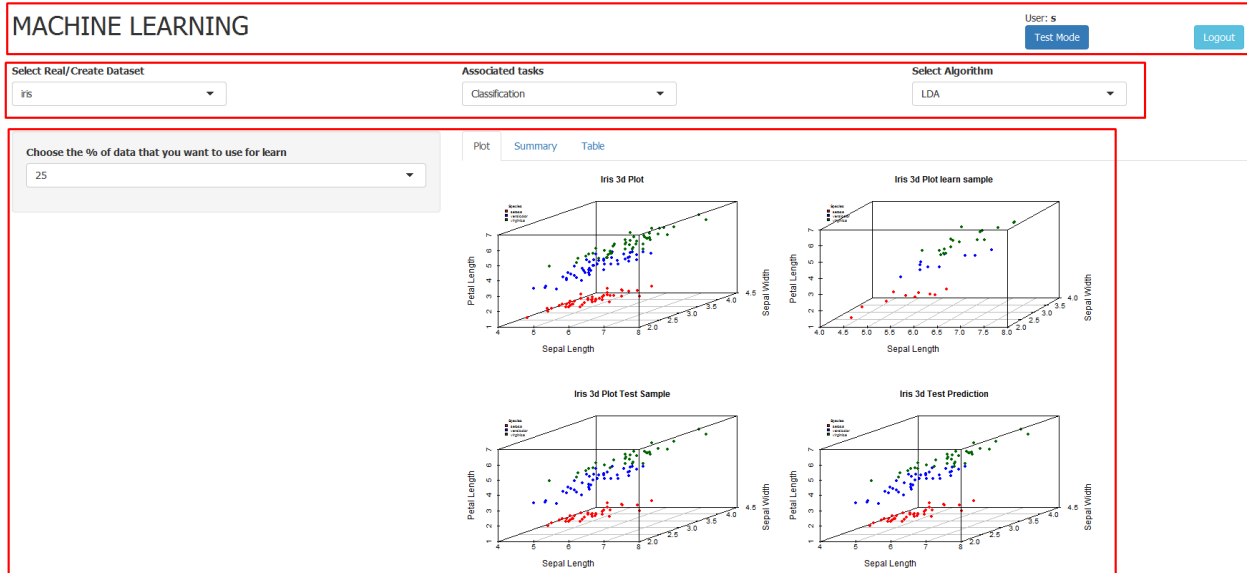


Figura 26: Estructura principal de la interfície part de learn

La primera part consisteix en un header on s'inclou el nom, l'usuari i dos botons per canviar a la part de test i l'altre per sortir de l'aplicació. Aquest header està present en les dues interfícies.

Com es pot observar a la imatge anterior, hem fet una selecció de dataset (iris) posteriorment hem escollit com a tasca a realitzar "Classificació" i per últim l'algoritme escollit ha estat el "LDA"

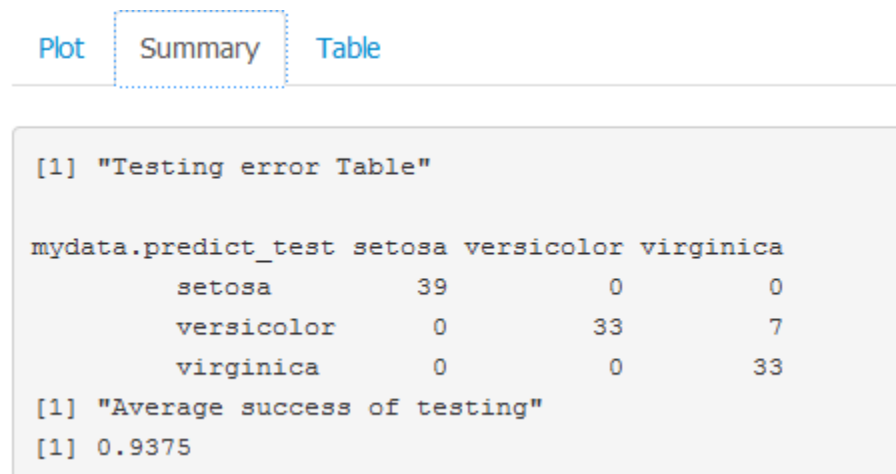
El temps de càrrega per mostrar els resultats del LDA amb els paràmetres per defecte ha sigut inferior a 3s.

Un cop tenim els resultats, podem variar la llista de paràmetres que es troba en el panell lateral per tal de recalculer l'algoritme en temps real.

Com es pot observar l'interacció amb l'usuari es molt fluida, pot intercanviar entre totes les pestanyes de forma fàcil i ràpida.

A cadascuna de les pestanyes tenim:

- PlotTab:
  - En aquesta pestanya, l'usuari podrà comprovar els resultats de l'algoritme , normalment en 4 "plots" diferents , un que serà el dataset original, un altre que serà les dades emprades per l'aprenentatge, un tercer consistent en les dades originals del test i per últim un "plot" amb les dades predites del dataset.
- SummaryTab:
  - En aquesta pestanya l'usuari podrà obtenir informació útil sobre l'algoritme, normalment obtindrà la matriu d'error del mètode i la "taxa d'encert"



The screenshot shows a web interface with three tabs: "Plot", "Summary", and "Table". The "Summary" tab is selected. Below the tabs, there is a code block containing the following text:

```
[1] "Testing error Table"

mydata.predict_test setosa versicolor virginica
  setosa           39           0           0
  versicolor       0           33           7
  virginica        0           0           33

[1] "Average success of testing"
[1] 0.9375
```

Figura 27: Pestanya Summary d'un algoritme.

- TableTab:
  - En aquesta pestanya podem comprovar els valor del dataset que hem escollit per realitzar l'algoritme.

Plot Summary **Table**

	Sepal.L.	Sepal.W.	Petal.L.	Petal.W.	Sp
1	5.10	3.50	1.40	0.20	setosa
2	4.90	3.00	1.40	0.20	setosa
3	4.70	3.20	1.30	0.20	setosa
4	4.60	3.10	1.50	0.20	setosa
5	5.00	3.60	1.40	0.20	setosa
6	5.40	3.90	1.70	0.40	setosa
7	4.60	3.40	1.40	0.30	setosa
8	5.00	3.40	1.50	0.20	setosa
9	4.40	2.90	1.40	0.20	setosa
10	4.90	3.10	1.50	0.10	setosa

Figura 28: Taula de dades (iris)

### 5.3 - Exemple d'execució de la part de test

En aquesta part mantenim el “header” explicat a la secció de la part de learn, modificant el botó que redirigeix a l'altre secció.

Aquesta és la pantalla principal de la part de test

#### Machine Learning Test

User: s

Learn Mode

Logout

Choose the number of questions

5

Choose the complexity of test

easy

START TEST

SHOW MY RESULTS

PLOT MY RESULTS

Figura 29: Interfície principal de la part de Test.

Al igual que la part principal de la secció de learn la pantalla principal del test la podem dividir en un panell lateral (sidebar) i el panell central on apareixeran, les preguntes, els resultats o els plots segons ho desitgi l'usuari.

En el panell lateral trobem dos llistes desplegable: la primera amb el nombre de preguntes que vol l'usuari que tingui el test, i la segona llista amb la dificultat d'aquest. Cal dir que la dificultat radica amb el nombre de respostes de la pregunta. En el cas d'un test de dificultat "easy" el nombre de respostes serà 2 i només una resposta serà correcte, en el cas d'un test de dificultat "medium" el nombre de respostes serà 3 i també d'una única resposta. I per últim en el cas d'un test de dificultat "hard" el nombre de respostes serà  $> 3$  i  $\leq 5$  i en aquest cas les respostes correctes poden ser des de 0 fins a la totalitat de respostes de la pregunta.

També tenim tres botons, el primer per iniciar el test i els altres per visualitzar el resultat (tant de plot com de text).

Choose the number of questions

5 ▼

Choose the complexity of test

easy ▼

START TEST SHOW MY RESULTS PLOT MY RESULTS

Figura 30: Panell lateral de la part de test

Si l'usuari prem Start test s'activa la funció que ens mostra les preguntes una a una i que aniran avançant quan l'usuari contesti i premi sobre el botó next. Aquest procés continua fins que acabi el test.

# Test of Machine Learning

## Pregunta 2

Please Select:

Respuesta 1

Respuesta 2

Next

Figura 31: Panell central de preguntes.

Un altre funcionalitat, s'activa al clicar "showmyresults" en la qual el panell condicional es mostrarà i ens ensenya els resultats obtinguts d'una consulta a la base de dades que explicarem en el funcionament intern d'aquesta part.

	dificultad	resultado	hora
1	easy	Total result of your test : 2/5	14-03-15 20:52:16
2	easy	Total result of your test : 5/5	14-03-15 21:07:58
3	easy	Total result of your test : 4/5	14-03-15 21:09:48
4	easy	Total result of your test : 3.5/5	14-03-15 21:09:53
5	easy	Total result of your test : 2/5	14-03-15 21:12:38
6	easy	Total result of your test : 2/5	14-03-15 21:12:44
7	medium	Total result of your test : 2/5	14-03-15 21:13:18
8	easy	Total result of your test : 10/10	03-04-15 17:42:27
9	easy	Total result of your test : 9.5/10	03-04-15 17:42:35
10	easy	Total result of your test : 3/15	03-04-15 18:12:33
11	easy	Total result of your test : 3/5	12-04-15 14:31:12
12	easy	Total result of your test : 3.5/5	12-04-15 14:31:20

Figura 32: Panell central de resultats.

Per acabar tenim un altre funcionalitat similar a l'anterior però aquesta ens mostrarà els resultats del tests en un barplot.

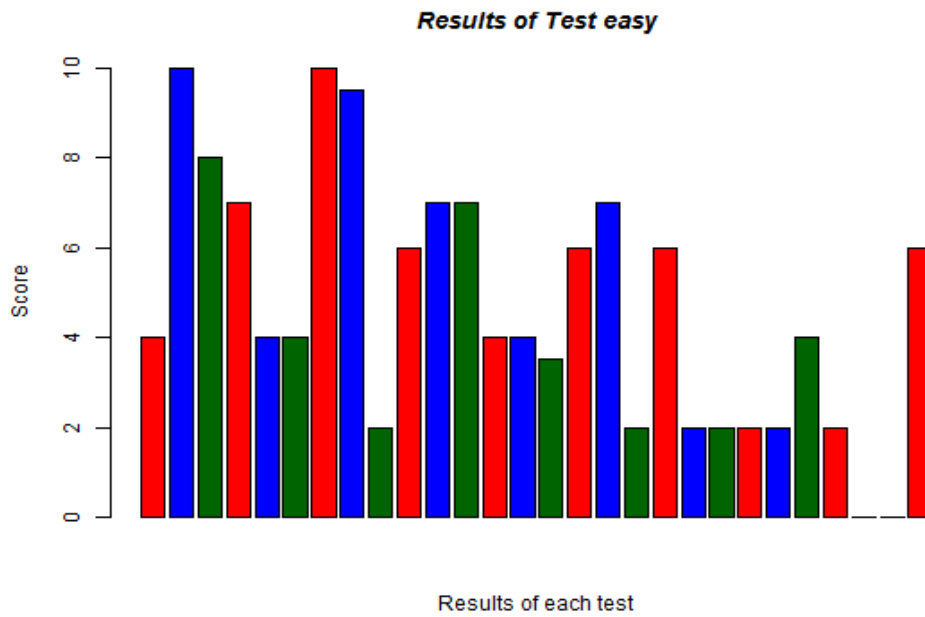


Figura 33: Panell central amb els resultats amb plot.

# TERCERA PART: MILLORES I CONCLUSIONS



# 1 - Millores Futures

Encara que en línies generals estem contents amb el desenvolupament del projecte, aquest té moltes vies de continuïtat i moltes millores que es poden fer per donar-li més riquesa i utilitat.

- Es pot implementar un sistema de “ludificació” per la part de test, consistent en l’adquisició de trofeus o “badges” per part de l’usuari en superar alguns reptes.
- Tan la part de test com la part de learn es poden millorar incorporant algoritmes o fins i tot fer que l’interfície sigui més amigable, com per exemple que l’usuari pugui seleccionar els eixos dels plots, que tingui la possibilita de girar els plots etc.
- Incorporar mètodes combinats per exemple un mètode de FeatureExtraction + Classificació.
- Establir que l’usuari pugui pujar al sistema els seus “datasets” i fer proves amb ells.
- Pujar l’aplicació al portal de science del departament de LSI (<http://science.lsi.upc.edu/>)

## 2 - Conclusions

Els objectius del projecte eren crear una interfície web que serveixi per aprendre *Machine Learning*, a més, aquesta aplicació hauria de ser útil i usable, així com facilitar el seguiment de l'assignatura als alumnes i finalment que l'usuari es pogués avaluar els seus coneixements, per tant l' interfície resultant d'aquest projecte resol de forma adequada encara que millorable els objectius proposat inicialment.

Aquesta és una interfície amigable, i amb la que l'usuari pot fiançar els seus coneixements de *Machine Learning*.

L' interfície que finalment tenim és molt similar a la que inicialment vam pensar, encara que amb petites modificacions per facilitar la feina a l'usuari, i que fan que l'usabilitat sigui millor.

Com es pot observar en l'apartat de "Milliores Futures", encara hi queda treball perquè la aplicació sigui del tot perfecte.

Durant el desenvolupament del projecte hem tingut alguns problemes com ara entendre com funcionava internament RShiny i les funcions reactives, per altra banda el sistema de "login" amb la base de dades va costar més de lo que inicialment pensava, per acabar, vam tenir un problema de planificació ja que al trobar un treball es va reduir notablement el temps diari per fer el TFG.

Finalment com a conclusions personals podem dir que aquest projecte ha sigut força interessant ja que m'ha proporcionat experiència en un camp que per mi el trobava atractiu, a més de la satisfacció de que aquesta aplicació serà utilitzada per usuaris finals, és a dir que veurà la llum, ja que probablement aquesta aplicació aparegui en el portal de science del departament de lsi ( <http://science.lsi.upc.edu/> ).

També ha sigut una experiència enriquidora tant a nivell acadèmic com professional, ja que es tracta d'un camp què està en ple creixement en el mercat actual, i he pogut aprendre i tenir nocions de noves eines.

# Bibliografia

- [1] *MACHINE LEARNING*, TOM MITCHELL, MCGRAW HILL, 1997
- [2] APRENTATGE AUTOMÀTIC - <http://www.lsi.upc.edu/~bejar/apren/docum/trans/00-introAprendizaje-eng.pdf> - Consultat a 05/10/2014
- [3] <http://cran.r-project.org/doc/contrib/rdebut.es.pdf> - Consultat a 06/10/2014
- [4] <http://www.r-project.org/> - Consultat a 06/10/2014
- [5] <http://www.rstudio.com/products/rstudio/> - Consultat a 06/10/2014
- [6] <https://github.com/rstudio/shiny/blob/master/README.md> - Consultat a 07/10/2014
- [7] <http://getbootstrap.com/> - Consultat a 22/10/2014
- [8] <http://shiny.rstudio.com/gallery/kmeans-example.html> - Consultat a 15/10/2014
- [9] <http://www.pmoinformatica.com/2012/10/plantillas-scrum-historias-de-usuario.html> - Consultat el 10/01/2015
- [10] *SCRUM EN 15 MINUTOS*. <http://es.slideshare.net/rodrigoiscrum-en-15-minutos> - Consultat a 06/10/2015
- [11] *MATLAB* - <http://es.mathworks.com/products/matlab/> - Consultat a 05/11/2014
- [12] ALPAYDIN, E , *INTRODUCTION TO MACHINE LEARNING* , MIT PRESS , 2010 , ISBN:026201243X. [http://cataleg.upc.edu/record=b1375315~S1\\*cat](http://cataleg.upc.edu/record=b1375315~S1*cat)
- [13] BISHOP, C.M , *PATTERN RECOGNITION AND MACHINE LEARNING* , SPRINGER , 2006 , ISBN:0387310738. [http://cataleg.upc.edu/record=b1298151~S1\\*cat](http://cataleg.upc.edu/record=b1298151~S1*cat)
- [14] CHERKASSKY, V.; MULIER, F , *LEARNING FROM DATA: CONCEPTS, THEORY, AND METHODS* , JOHN WILEY , 2007 , ISBN:0471681822. [http://cataleg.upc.edu/record=b1352176~S1\\*cat](http://cataleg.upc.edu/record=b1352176~S1*cat)
- [15] HAYKIN, S.S , *NEURAL NETWORKS AND LEARNING MACHINES* , PRENTICE HALL , 2009 , ISBN:0131471392. [http://cataleg.upc.edu/record=b1342098~S1\\*cat](http://cataleg.upc.edu/record=b1342098~S1*cat)

[16] MITCHELL, T.M , *MACHINE LEARNING* , THE MCGRAW-HILL COMPANIES , 1997 , ISBN:0070428077 .  
[http://cataleg.upc.edu/record=b1133468~S1\\*cat](http://cataleg.upc.edu/record=b1133468~S1*cat)

[17] FEATURE EXTRACTION - [http://en.wikipedia.org/wiki/Feature\\_extraction](http://en.wikipedia.org/wiki/Feature_extraction) - Consultat a 5/03/2015

[18] CLASSIFICACIÓ - <http://www-users.cs.umn.edu/~kumar/dmbook/ch4.pdf> – Consultat a 8/03/2015

[19] REGRESIÓ [http://en.wikipedia.org/wiki/Regression\\_analysis](http://en.wikipedia.org/wiki/Regression_analysis) - Consultat a 09/03/2015 - Consultat a 12/03/2015

[20] CLUSTERING – <http://www.cs.upc.edu/~bejar/apren/docum/trans/08-clustering-eng.pdf> - Consultat a 15/03/2015

[21] CLUSTERING - [http://en.wikipedia.org/wiki/Cluster\\_analysis](http://en.wikipedia.org/wiki/Cluster_analysis) - Consultat a 15/03/2015

# Glossari

(1) - **Funció Reactiva** → Es diu d'una funció que es cridada automàticament des de el client al haver-hi una interacció amb un element de la interfície. Podent d'aquesta forma generar resultats en temps real per a cada "input" de l'usuari.

(1.1) - **Input** → Acció i valor d'un camp de l'aplicació necessari per l'algoritme.

(2) **Mòdul**→ Especificuem mòdul , com una classe de programació en el llenguatge "R"

(3) - **Aprenentatge supervisat** → Es diu que un problema és d'aprenentatge supervisat ja que a partir d'una mostra d'entrenament ("learn") on es coneixen els valors  $(x,y)$  es construeix una funció que permet predir el valor desconegut de "y" per a una nova observació de "x".

(4) - **dataset** → Conjunt de dades d'entrada dels nostres algoritmes.

(5) - **aprenentatge no supervisat** → En l'aprenentatge no supervisat , a diferència del supervisat, es disposa únicament de valors de "x" sense conèixer cap valor de "y".

# Annexes

- Manual d'usuari
- Manual d'administrador.
- Diagrama de Gantt Modificat

# USER MANUAL

This is the user manual to understand the flow of application and to have the basics for starting to use the application in the correct way.

*Web interface to  
learn machine  
learning*

# INDEX

---

1 - LOGGING	2
2 - HEADER	3
3 - LEARN PART	4
4 - TEST PART	9

# Logging to application

---

## 1 - Logging

To start using the application first of all you need to login with your credentials (These credentials should be given to you by your teacher).

When you open the application, the login layout will popup to you, please complete the fields to login.

The screenshot shows a login form with the following elements:

- The word "Login" is centered at the top.
- Below it is the label "Account".
- A text input field containing the text "sonia.sandalina:" is highlighted with a red box and labeled with a red "1" to its right.
- Below that is the label "Password".
- A password input field with ten dots and a cursor is highlighted with a red box and labeled with a red "2" to its right.
- At the bottom is a green button with the text "Login" in white.

1. Here you need to type your username
2. Here you need to type your password



## LEARN AND TEST PARTS

---

### 2 - HEADER

If credentials are correct ,the system will redirect you to the first page of application.

MACHINE LEARNING



Here is the **header** of application where you can see your username (1) , you can change between learn/test mode(2), and you have the option to logout(3) from application.

## 3 - LEARN PART

Now we start to explain how to use the **learn** part.

- 1) First you need to select between Real datasets (iris, decathlon, tea, hobbies) or created datasets by system (parabolic, Fibonacci)

### Select Real/Create Dataset

### Select Real/Create Dataset

- Select Dataset
- iris
- decathlon
- parabolic
- fibonacci
- tea
- hobbies

- 2) When you select one of this datasets another selected list will appear at the right . In this you need to select what associated task to do with the dataset selected (Classification, Regression, Feature Extraction or Clustering). **Not all datasets are valid to do all tasks.**

### Associated tasks

## Associated tasks

Select Type

- Select Type
- Classification
- Regression
- Feature Extraction
- Clustering

3) Depending on associated task selected, another selected field will appear at the right, at this moment you will select the algorithm to perform with dataset selected.

a. For example : Iris → Classification → LDA

Select Real/Create Dataset: iris

Associated tasks: Classification

Select Algorithm: LDA

As we said, depending on associated task, algorithms can vary.

- If we select Classification :
  - LDA
  - QDA
  - NBAYES
  - MLP
  - SVM

Associated tasks: Classification

Select Algorithm: LDA

- If we select Regression :
  - MLP
  - SVM

# USER MANUAL

Associated tasks

Regression

Select Algorithm

Select|

Select

MLP

SVM

- If we select Feature Extraction
  - PCA
  - LDA
  - MCA (Only “Tea”, “Hobbies”)

Associated tasks

Feature Extraction

Select Algorithm

Select|

Select

PCA

LDA

Associated tasks

Feature Extraction

For "Tea" and "Hobbies" Datasets

Select Algorithm

Select|

Select

MCA

- If we select Clustering
  - KMEANS

Associated tasks

Clustering

Select Algorithm

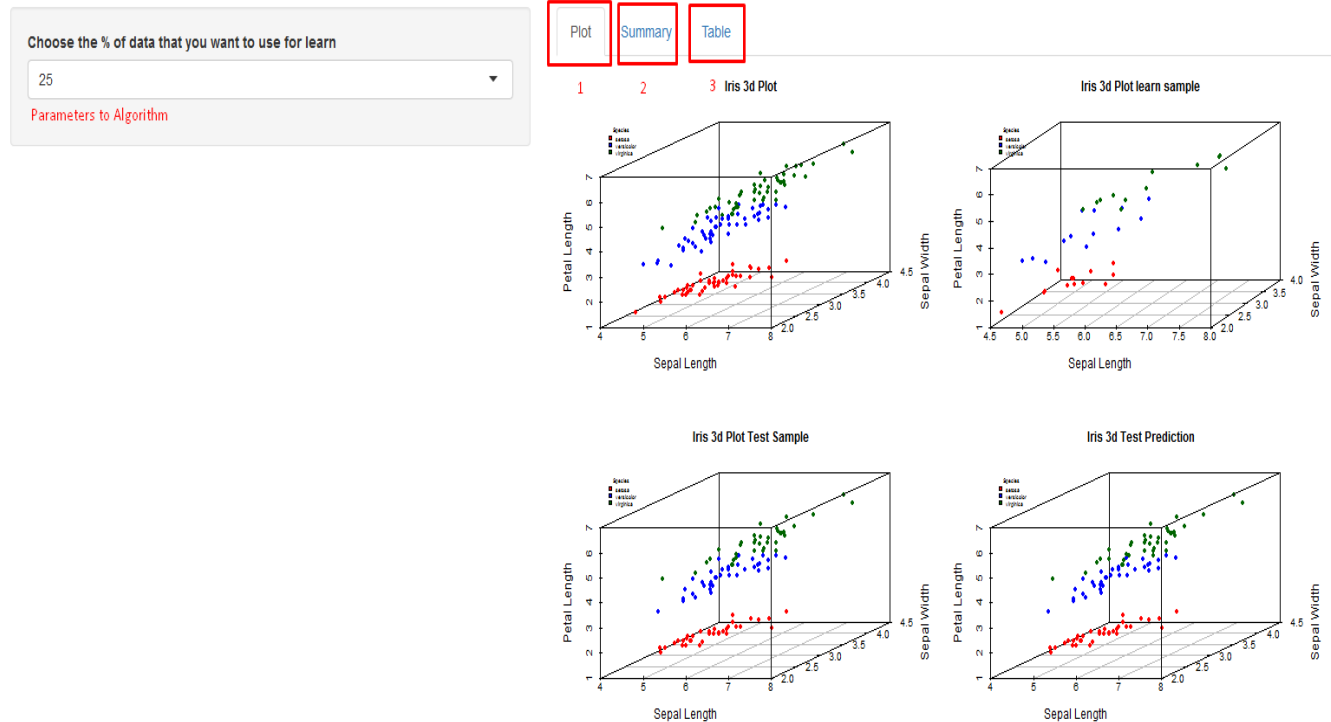
KMEANS

So, now, we going to explain how to change parameters and how to visualize results.

Imagine for example, that we select “Iris” , “Classification” , and “LDA” as algorithm.

When we click to LDA, a new panel will be rendered, see the next image.

# USER MANUAL



Mostly, you can change parameters to visualize how the parameters affect to our dataset.

You have 3 tabs:

- 1) The first tab is where you can see the result of the algorithm by graphics.
- 2) The second tab, summary, is where you can see the results of the algorithm in text format.
- 3) The last tab, "Table", is where you can see the entire dataset.

If you change the parameter to another value( for example 50%), the graphics and the rest of tabs will upload with the new result.

All tabs are clickable, and you can change between them without losing the results.

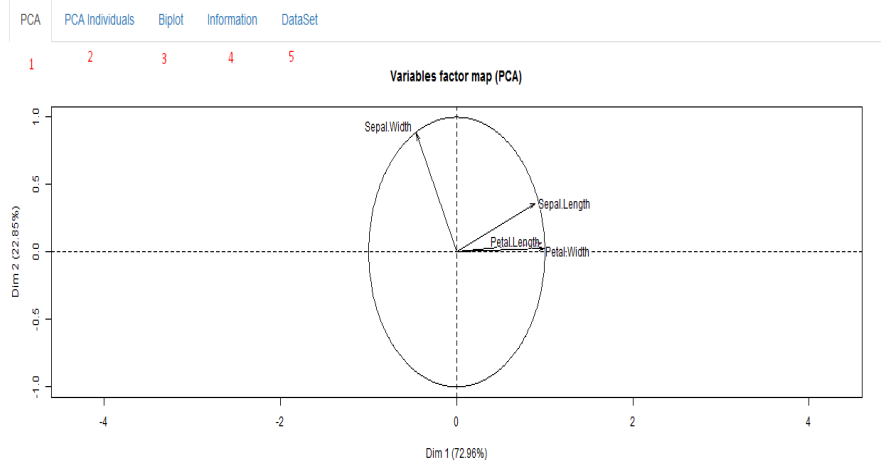
Another example :

We are going to select "iris" , "feature extraction" and "PCA"

As we can see in the next image, is a little different than LDA from classification, this is because we need to show you different results in different plots, but the usability is the same, you can change between tabs as you want.

# USER MANUAL

This method is used to find the causes of variability in a dataset and sort them by importance



One , two and Third tab, are graphics to see how PCA resolve the dataset, the fourth tab is a summary of the result and the fifth tab is the same as “table” , the dataset printed.

## 4 - TEST PART

Now , we start to explain the **test** mode

First of all, to enter to “test” mode, you need to click the button at top right , see the image below.

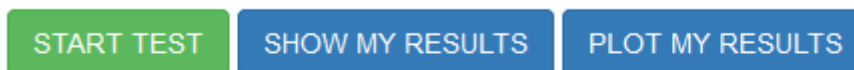


When you click to this button you will be redirected to the test layout.

## Machine Learning Test

Choose the number of questions

Choose the complexity of test



Here you can perform different tests, to evaluate your knowledge about machine learning.

At left you can see two selected fields, one of them are the number of questions of the test (**5,10,15**) , and the other is the complexity ( **easy, medium, hard** )

As you can see, there are three buttons.

- **START TEST** : If you click in this button, you start a new test with the parameters selected

# USER MANUAL

---

Choose the number of questions  
5

Choose the complexity of test  
easy

**START TEST** **SHOW MY RESULTS** **PLOT MY RESULTS**

## Test of Machine Learning

Welcome to Machine Learning Test

Next

Now you need to push “next” to start the test, and first question will appear, when you answer this question click another time to next, when the test is finished, your results will be shown

## Test of Machine Learning

### Pregunta 1

Please Select:

- Respuesta 1
- Respuesta 2

Next

## Test of Machine Learning

### Your results

```
[1] "Question 1 has: CORRECT ANSWER"  
[1] "Question 2 has: CORRECT ANSWER"  
[1] "Question 3 has: CORRECT ANSWER"  
[1] "Question 4 has: INCORRECT ANSWER"  
[1] "Question 5 has: INCORRECT ANSWER"  
[1] "Total result of your test : 3/5"
```

Next

**SHOW MY RESULTS** : If you click in this button, all your tests will be shown at the right.



# USER MANUAL

Choose the number of questions  
5

Choose the complexity of test  
hard

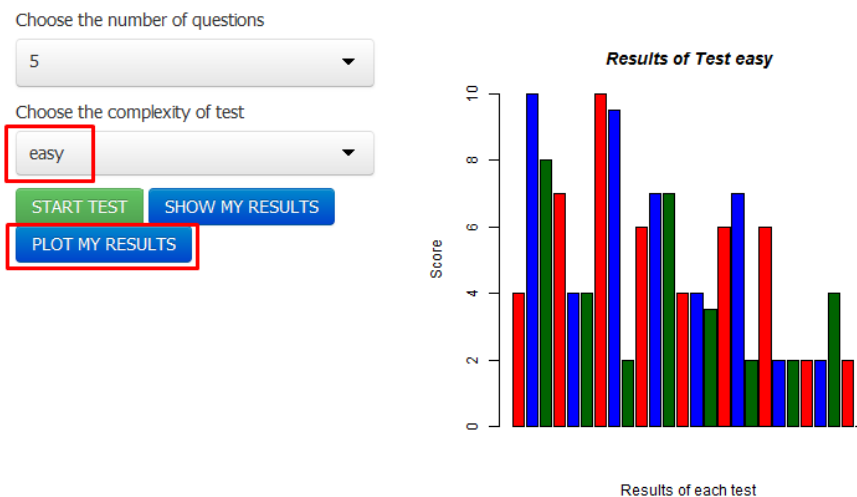
START TEST SHOW MY RESULTS PLOT MY RESULTS

	dificultad	resultado	hora
1	easy	Total result of your test : 2/5	14-03-15 20:52:16
2	easy	Total result of your test : 5/5	14-03-15 21:07:58
3	easy	Total result of your test : 4/5	14-03-15 21:09:48
4	easy	Total result of your test : 3.5/5	14-03-15 21:09:53
5	easy	Total result of your test : 2/5	14-03-15 21:12:38
6	easy	Total result of your test : 2/5	14-03-15 21:12:44
7	medium	Total result of your test : 2/5	14-03-15 21:13:18
8	easy	Total result of your test : 10/10	03-04-15 17:42:27
9	easy	Total result of your test : 9.5/10	03-04-15 17:42:35
10	easy	Total result of your test : 3/15	03-04-15 18:12:33
11	easy	Total result of your test : 3/5	12-04-15 14:31:12
12	easy	Total result of your test : 3.5/5	12-04-15 14:31:20
13	easy	Total result of your test : 3.5/5	12-04-15 14:31:24
14	easy	Total result of your test : 4/10	12-04-15 14:31:43
15	easy	Total result of your test : 6/15	12-04-15 14:31:57
16	easy	Total result of your test : 3.5/10	12-04-15 14:32:06

As you can see in the image, all of your results, with complexity, result and date will show when you press “Show my results”

- PLOT MY RESULTS : Depending on complexity that are selected, a plot with total of results will appear

If you press in “plot my results” with complexity “easy” or “medium” or “hard” a plot with all of your tests that satisfy the criteria will appear.



# USER MANUAL

---

Choose the number of questions

5

Choose the complexity of test

medium

START TEST SHOW MY RESULTS

PLOT MY RESULTS



Choose the number of questions

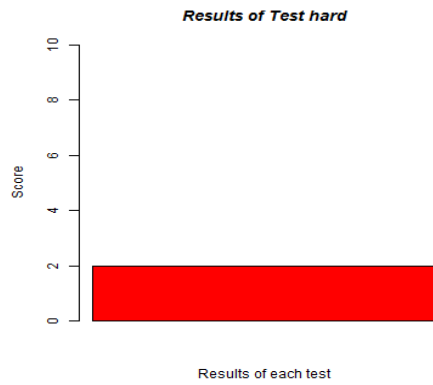
5

Choose the complexity of test

hard

START TEST SHOW MY RESULTS

PLOT MY RESULTS



# ADMIN MANUAL

This manual will help you to create a database for the web application :  
"Aprendre Aprenentatge"

*How to Create  
Database and  
upload questions  
and tests*

# INDEX

---

<b>1 – MYSQL</b>	<b>2</b>
<b>2 – WORKBENCH</b>	<b>2</b>
<b>3 – WORKBENCH INSTANCE</b>	<b>3</b>
<b>4 - SCHEMA</b>	<b>5</b>
CREATE SCHEMA QUERY	5
<b>5 – TABLES</b>	<b>5</b>
CREATE TABLE QUERIES.	5
CREATE TEST REGISTER	7
CREATE GRUPOPREGUNTAS REGISTER	7
CREATE PREGUNTAS REGISTER	8
CREATE RESPUESTAS REGISTER	8
<b>CREATE USERS</b>	<b>9</b>

# REQUIREMENTS

---

## 1 - MYSQL

First of all , you need to download the last version of MySQL.

You can download here : <http://www.mysql.com/downloads/>

In order to facilitate the process to integrate with web application, we suggest that you should create a user “root” with password “1234”.

## 2 - WorkBench

We strongly recommend to use MySQL Workbench or PhpMyAdmin or something similar to create the schema, in this manual we explain how to create through text queries in MySQL Workbench 6.2.

Download link : <https://dev.mysql.com/downloads/workbench/>

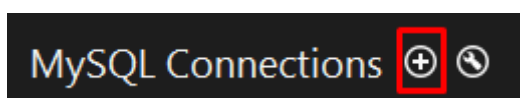
# CREATING THE INSTANCE

## 3 – Workbench Instance

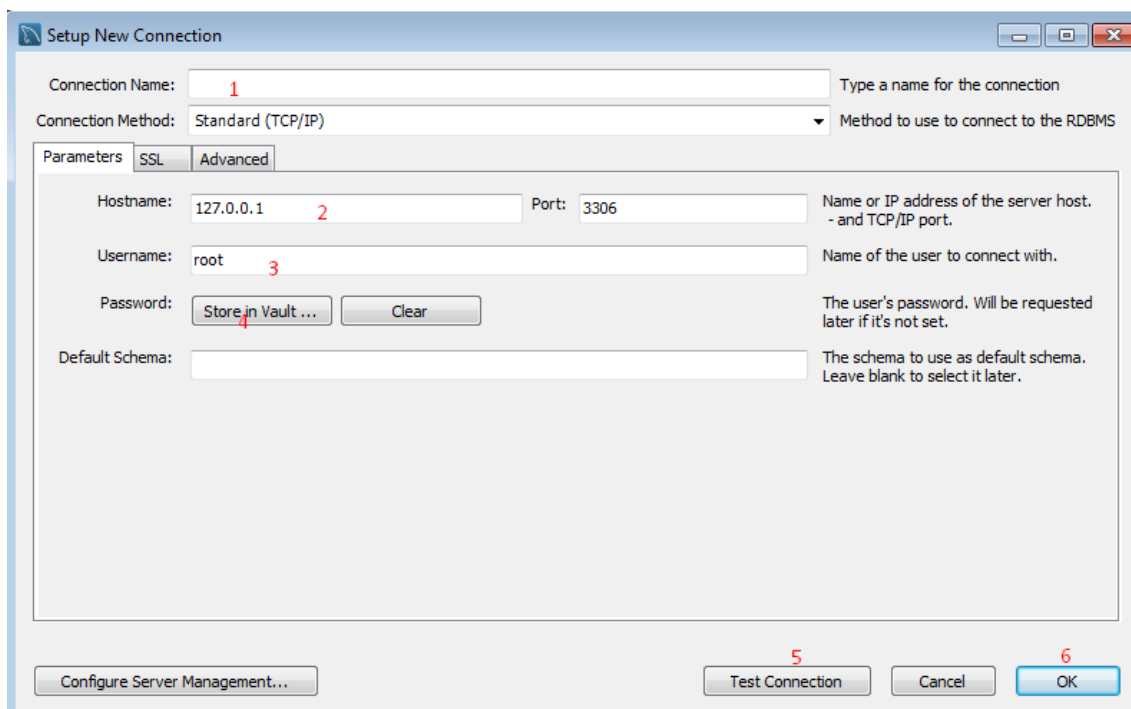
In order to create schema, please open your “MySQL Workbench”



When opened, please add a new connection.



When you click in add new connection, another window will appear.



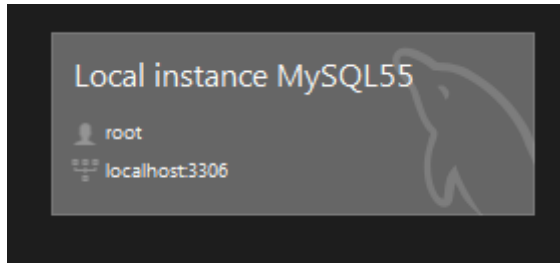
Here you can add your new connection:

1. Connection name : A name that helps you to identify instance, for example “LOCAL INSTANCE”
2. Hostname : If you are in development, 127.0.0.1 is perfect, if you want to point to some server, here you need to write the public ip address.
3. Username : As we said in requirements, we recommend you to assign this value to “root”.

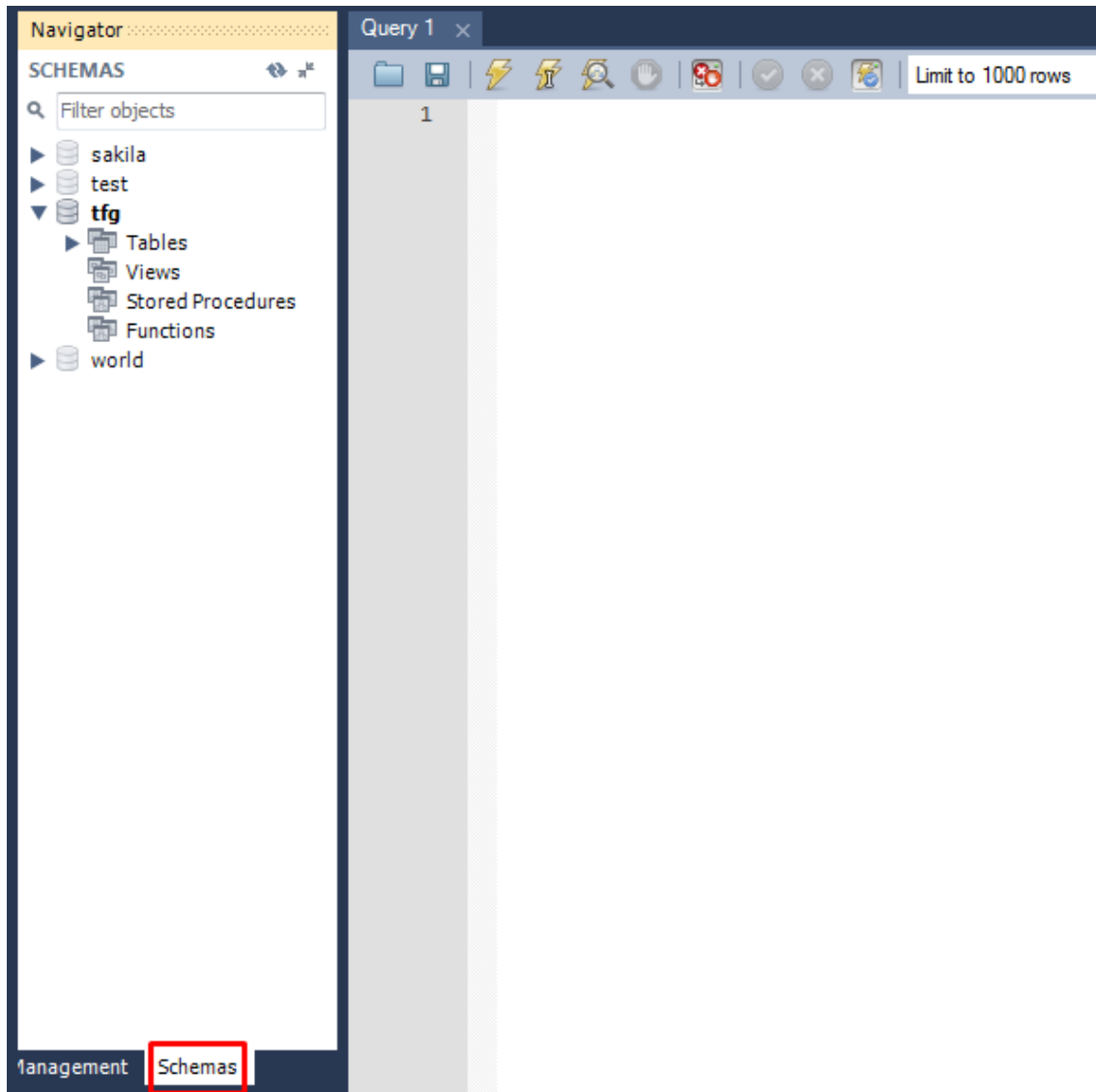
## Admin MANUAL

---

4. When you press “Store in Vault...” a new window when you need to write the password (1234) appear.
5. Finally , test your connection, is everything is ok go to point 6
6. Save connection and use it.



Now, you need to log in our instance with double click and then, another window will appear.



## Creating the SCHEMA

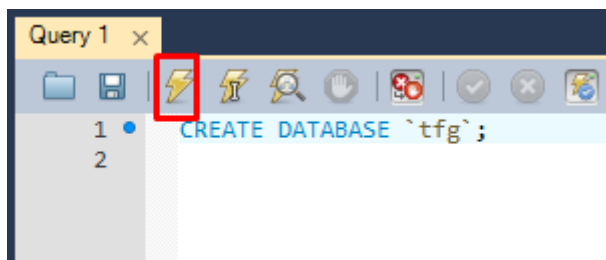
---

### 4 - Schema

Now that you are logged to mysql workbench, you need to create schema for this please write the next sentence in right place of your workbench and press the beam icon.

#### Create schema query

```
CREATE DATABASE `tfg`;
```



### 5 - Tables

Now, we're going to create all needed tables.

#### Create table queries.

```
CREATE TABLE `usuarios` (  
  `idusuarios` int(11) NOT NULL AUTO_INCREMENT,  
  `correo` varchar(220) NOT NULL,  
  `password` varchar(45) NOT NULL,  
  PRIMARY KEY (`idusuarios`),  
  UNIQUE KEY `idusuarios_UNIQUE` (`idusuarios`)  
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `grupopreguntas` (  
  `idgrupoPreguntas` int(11) NOT NULL,  
  `idTest` varchar(45) NOT NULL,  
  `idPreguntas` varchar(250) NOT NULL,  
  PRIMARY KEY (`idgrupoPreguntas`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```



# Admin MANUAL

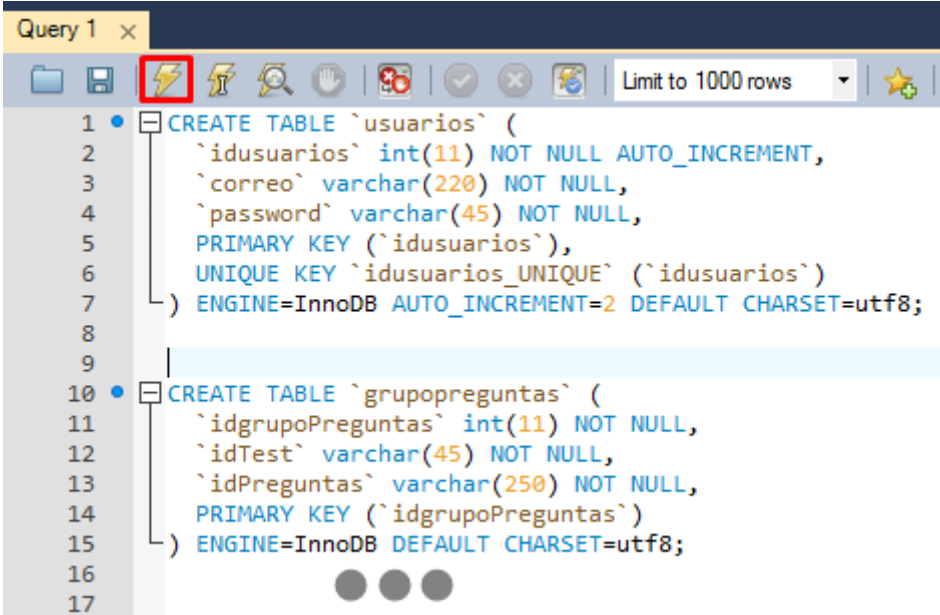
---

```
CREATE TABLE `preguntas` (  
  `idPregunta` int(11) NOT NULL,  
  `respuestasCorrectas` varchar(220) DEFAULT NULL,  
  `respuestasIncorrectas` varchar(220) DEFAULT NULL,  
  `pregunta` varchar(500) NOT NULL,  
  PRIMARY KEY (`idPregunta`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `respuestas` (  
  `idRespuesta` int(11) NOT NULL,  
  `respuesta` varchar(500) NOT NULL,  
  PRIMARY KEY (`idRespuesta`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `test` (  
  `idTest` int(11) NOT NULL AUTO_INCREMENT,  
  `dificultad` varchar(45) NOT NULL,  
  `numeroPreguntas` int(11) NOT NULL,  
  PRIMARY KEY (`idTest`)  
) ENGINE=InnoDB AUTO_INCREMENT=11 DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `testusuario` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `correo` varchar(250) NOT NULL,  
  `dificultad` varchar(250) NOT NULL,  
  `resultado` varchar(250) NOT NULL,  
  `hora` varchar(200) NOT NULL,  
  PRIMARY KEY (`id`),  
  KEY `usuarioId_idx` (`correo`),  
  KEY `testId_idx` (`dificultad`),  
  KEY `correo_index` (`correo`) USING BTREE  
) ENGINE=InnoDB AUTO_INCREMENT=55 DEFAULT CHARSET=utf8;
```



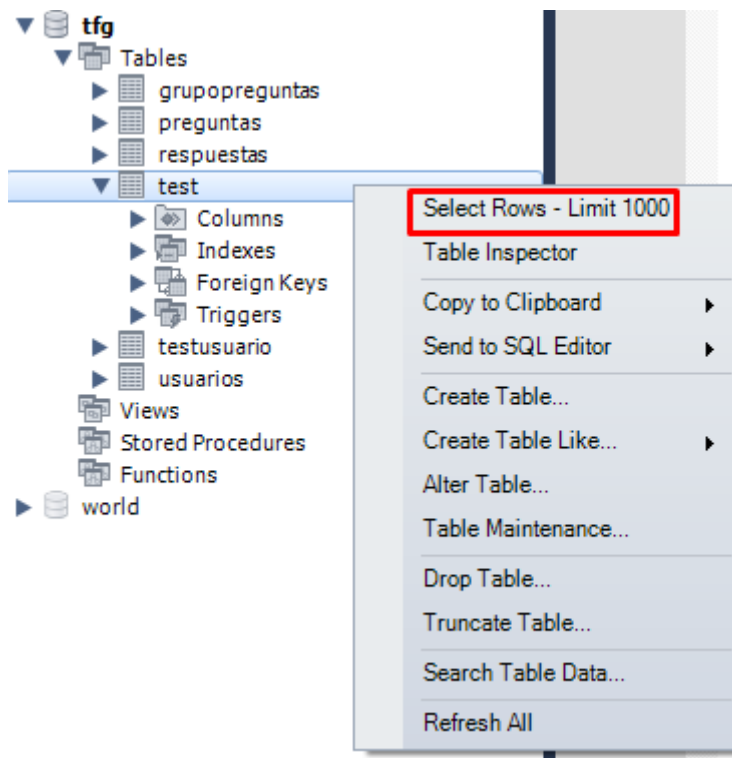
The screenshot shows a MySQL query editor window titled "Query 1". The toolbar includes icons for file operations, a lightning bolt icon (highlighted with a red box), search, and other standard tools. A dropdown menu is set to "Limit to 1000 rows". The SQL code is as follows:

```
1 • CREATE TABLE `usuarios` (  
2   `idusuarios` int(11) NOT NULL AUTO_INCREMENT,  
3   `correo` varchar(220) NOT NULL,  
4   `password` varchar(45) NOT NULL,  
5   PRIMARY KEY (`idusuarios`),  
6   UNIQUE KEY `idusuarios_UNIQUE` (`idusuarios`)  
7 ) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8;  
8  
9  
10 • CREATE TABLE `grupopreguntas` (  
11   `idgrupoPreguntas` int(11) NOT NULL,  
12   `idTest` varchar(45) NOT NULL,  
13   `idPreguntas` varchar(250) NOT NULL,  
14   PRIMARY KEY (`idgrupoPreguntas`)  
15 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;  
16  
17
```

# CREATING TESTS AND QUESTIONS

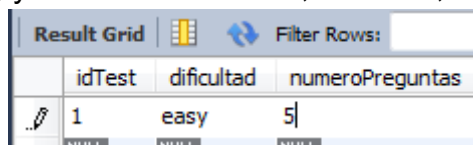
Now we start to explain how to use the create 1 Test and assign them some questions and answers.

For this example we suppose we want an “easy” test with 5 questions.



## Create Test register

- First, you need to right click on “test” and select “Select rows...”
- Then , you need to write “id,dificultad,numeropreguntas” on right panel



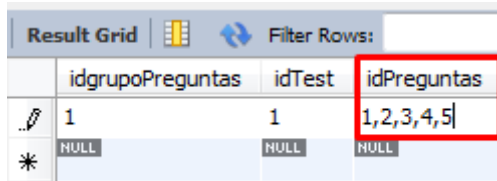
- And to save this, press “Apply” in bottom right



## Create GrupoPreguntas register

## Admin MANUAL

- For insert a new “grupopreguntas”, you need to do the same as “test” , the next image show an example of input.

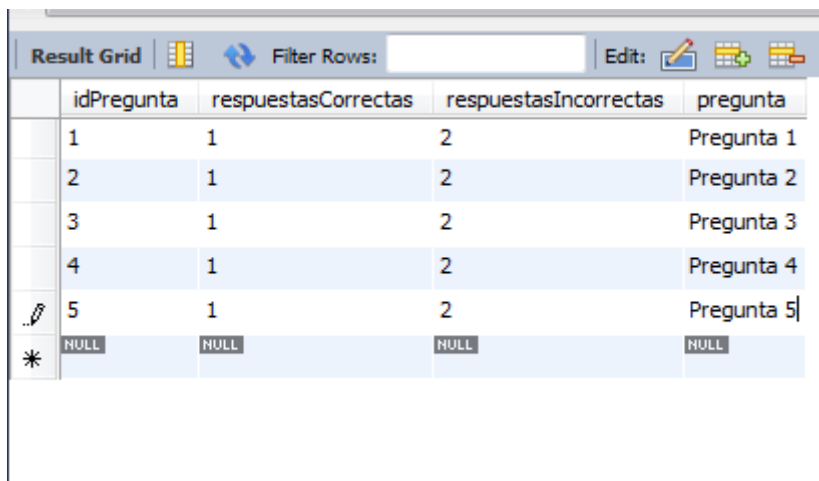


	idgrupoPreguntas	idTest	idPreguntas
✎	1	1	1,2,3,4,5
*	NULL	NULL	NULL

“idPreguntas” is a string in Comma separated format of id’s of questions in table “preguntas”.

### Create Preguntas register

- For create a “Preguntas” register , you need to do the same, but here, you need to insert five register, one per question, and id’s of question should be the same as “idPreguntas” in table “grupoPreguntas”, in the next image you will se an example.



	idPregunta	respuestasCorrectas	respuestasIncorrectas	pregunta
	1	1	2	Pregunta 1
	2	1	2	Pregunta 2
	3	1	2	Pregunta 3
	4	1	2	Pregunta 4
✎	5	1	2	Pregunta 5
*	NULL	NULL	NULL	NULL

### Create Respuestas register

- And for finish, we need to complete the table “Respuestas” with the text of answer for each id in “preguntas.respuestasCorrectas” and “preguntas.respuestasIncorrectas”. See the next image for example.

Result Grid		Filter Rows:
	idRespuesta	respuesta
	1	Respuesta 1
	2	Respuesta 2
*	NULL	NULL

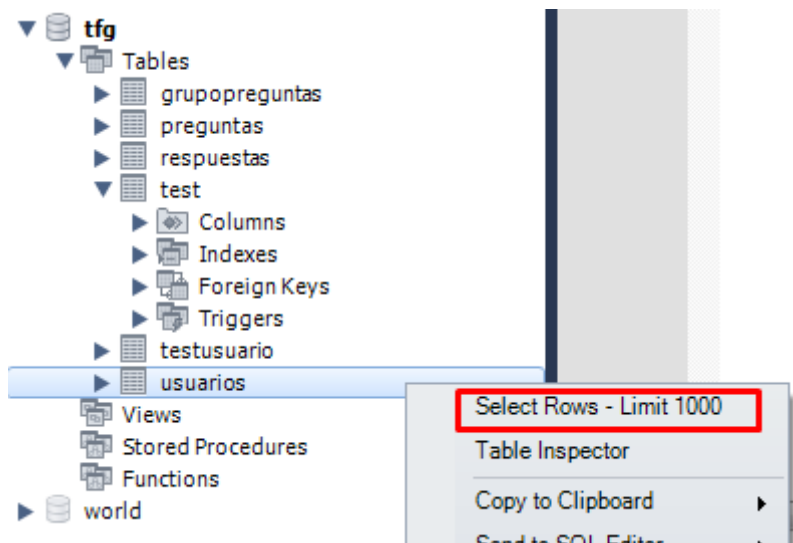
When all this inserts are finished, you can go to the application and perform a “easy” test with 5 questions.

## CREATE USERS

---

### Create users

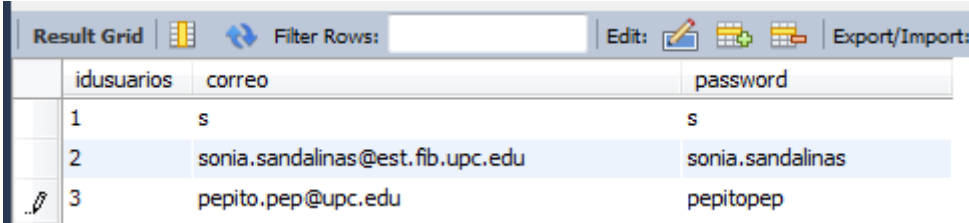
If you want to create users to grant acces to web application, you need to go to “usuarios” table.



# Admin MANUAL

---

Right click in this table and choose “Select Rows ...” , On the bottom right panel you can insert all the records that you want. See the next image for example



	idusuarios	correo	password
1		s	s
2		sonia.sandalinas@est.fib.upc.edu	sonia.sandalinas
3		pepito.pep@upc.edu	pepitopep

When you finish to add register, don't forget to press “Apply” button to save all.

# Gantt Modificat

