



UNIVERSIDAD COMPLUTENSE DE MADRID

PROYECTO DE SISTEMAS INFORMÁTICOS

Facultad de Informática

Curso 2012/2013

**Desarrollo de técnicas de detección automática de
estado de ánimo mediante redes de sensores inalámbricas**

Autores:

Miguel CISNEROS
Alejandro MALAGON
José Pablo REDONDO

Director:

Román Hermida
José Luis Ayala

PREFACIO

La detección de emociones es un campo de investigación en el cual queda mucho por descubrir. En este área, se plantea la captura y clasificación de las señales eléctricas biométricas de los sujetos como mecanismo de detección automática de emociones.

A continuación expondremos un proceso para la inducción de emociones y la captura y tratamiento de las señales biométricas generadas por el sujeto.

Con los datos proporcionados realizaremos un estudio para averiguar si existe algún tipo de correlación entre los datos. Mediante este estudio intentaremos averiguar si es posible saber en qué estado anímico se encuentra un individuo solamente por sus variables biométricas.

Keywords: móvil, servidor, emociones, estadística, inteligencia artificial, som, estudio.

AUTORIZACIÓN

Miguel Cisneros, Alejandro Malagón y José Pablo Redondo autorizan a la Universidad Complutense a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la propia memoria, como el código, la documentación y/o el prototipo desarrollado

A nuestros amigos, que aún con miedo a las agujas han cedido a las pruebas
y a todos aquellos que han estado a nuestro lado
para ayudarnos.

Nos gustaría agradecer la ayuda prestada por el profesor José Luis Valencia
Delfa (Facultad de estudios estadísticos U.C.M.) a la hora de ayudarnos durante
nuestros análisis iniciales para elegir correctamente los algoritmos.

1. INTRODUCCIÓN	11
1.1. ¿En qué consiste?	11
1.2. Motivación	11
1.3. Objetivos	12
1.4. Estudios Anteriores:	12
1.4.1. Emotion Recognition from Psychological Signals for User Modeling of Affect:	12
1.4.2. Emotion recognition using physiological signal from multiples subjects:	13
1.4.3. Emotion recognition using multilayer perceptron and generalized feed forward neural network:	14
1.4.4. Lifestyle management using wearable computer:	14
1.4.5. Activity-aware Mental Stress Detection Using Physiological Sensors.	15
1.4.6. Emotion Research.	15
1.4.7 Physionet.	16
1.4.8 Otros proyectos.	16
1.5. Visión general del documento, estructura.	17
2. Diseño y Especificación	19
2.1. Introducción	19
2.2. Metodología	19
2.2.1. Flujo completo de información	19
2.2.2. Hipótesis de partida (Emociones con impacto biométrico)	21
2.2.3. Captura de variables biométricas e inducción de estados anímicos.	21
2.2.4. Desarrollo de algoritmos de clusterización de señales	23
2.2.5. Evaluación experimental con síntesis emocional	23
2.3. Requisitos	24
2.3.1. Requisitos funcionales	24
2.3.2. Requisitos no funcionales	25
2.4. Investigación preliminar	26
2.4.1 Modos de inducir emociones	26
2.4.2. Métodos estadísticos estudiados	30
2.4.3. Distintos algoritmos de clasificación, librerías	34
2.4.4. ¿Dónde realizar el proyecto? SOs, Eclipse, etc	34
2.4.5. Algoritmos usados	35
3. Arquitectura	39
3.1. Introducción	39
3.2. Controlador	39
3.3. Captura y procesado de datos	40

3.4. SOM	41
3.5. Envío y recepción de datos al servidor	42
3.6. Tratamiento de datos en el servidor	42
3.7. Interfaz gráfica	43
4. PROCESO DE EXPERIMENTACIÓN	45
4.1. EVALUACIÓN EXPERIMENTAL DE LA SOLUCIÓN ADOPTADA	45
4.2. DESCRIPCIÓN DE LA MUESTRA EXPERIMENTAL (SUJETOS)	58
4.3. PROCESO DE EXPERIMENTACIÓN Y RESULTADOS OBTENIDOS	58
4.4. ANÁLISIS DE LA ESPECIFICIDAD Y EFICACIA DE LA SOLUCIÓN ADOPTADA	60
5. CONCLUSIONES	63
5.1. Usos de la aplicación	64
5.2. Futuras ampliaciones	65
5.3. Seguimiento del proyecto	66
6. GUIA DE USO	69
6.1. Introducción.	69
6.2. Requisitos	69
6.3. Instalación de la APK	70
6.4. Uso de la aplicación móvil	70
6.5. Instalación y uso del script para descargar datos	73
6.6. Instalación y montaje de los servidores	74
6.7. Información sobre las frases, videos, etc, para la inducción de emociones	82
7. TECNOLOGÍAS USADAS	83
8. APÉNDICE	89

1. INTRODUCCIÓN

1.1. ¿En qué consiste?

Este proyecto “Desarrollo de técnicas de detección automática de estado de ánimo en redes de sensores inalámbricas” consiste en la realización de un estudio científico sobre la detección y la estimación de algunas de las emociones más básicas del ser humano mediante el análisis de bio-señales tales como el pulso sanguíneo y la reacción Galvánica de la piel. Para ello, se utilizarán metodologías de Machine Learning sobre hardware móvil, de tal manera que sea posible realizarlo de manera completamente portátil.

1.2. Motivación

La detección automática de estados anímicos abre un campo de intensa investigación relacionada con las interfaces hombre-máquina y el campo biomédico. En particular, la detección de emociones puede permitir los siguientes avances:

- Interacción hombre-máquina basada en emociones, adecuación de mensajes computacionales al estado del usuario, detección de situaciones de estrés en conducción, adecuación dinámica del entorno en videojuegos.
- Desarrollo de terapias cognitivas y farmacológicas más efectivas para el tratamiento de desórdenes anímicos (depresión, paranoia, fobia, personalidad histriónica, etc.).

1.3. Objetivos

El objetivo del proyecto es la obtención de conclusiones demostrables científicamente sobre los siguientes aspectos:

- La relación intrínseca de las principales bio-señales humanas sobre las emociones. De tal forma que ellas, por sí solas, son suficientes para detectar la emoción en particular.
- La capacidad de los dispositivos móviles para sostener y procesar un algoritmo de alto consumo como la lectura de una gran fuente de datos y procesado sobre una red neuronal integrada en el mismo.
- La efectividad de pruebas empíricas para la inducción de emociones basadas en estudios previos y en nuestras propias investigaciones.
- La viabilidad de algoritmos de Machine Learning no supervisados para un problema típicamente resuelto con pruebas supervisadas.

1.4. Estudios Anteriores:

Ha habido numerosos estudios en el pasado acerca de la detección de las emociones, su tratamiento empírico y su reconocimiento basado en las bio-señales, escogiendo algunos de los más modernos de ellos, nosotros hemos escogido mejorar algunos de los aspectos de esas investigaciones, al tiempo que combinamos las técnicas de varios de ellos en un proyecto más completo.

1.4.1. Emotion Recognition from Psychological Signals for User Modeling of Affect¹:

¹ [Fatma Nasoz, Christine L. Lisetti, Kaye Alvarez, Neal Finkelstein. \(PONER AÑO\)](#)

Este estudio está basado en la utilización de señales fisiológicas, así como la detección de expresiones para el reconocimiento de una emoción específica. Muchas de las ideas de nuestro estudio se han obtenido partiendo de este mismo documento. Según el proyecto de la Dra. Nasoz, ellos utilizaron para la detección de emociones el dispositivo BodyMedia SenseWear. Parte de la detección de las señales que nosotros utilizamos procederán de este mismo dispositivo.

Otra de las aproximaciones de este documento fue el pre-procesamiento de las señales mediante una función de normalización. En este caso, nosotros mejoramos el algoritmo de normalización utilizando un algoritmo más estadístico.

Por último, la metodología de entrenamiento que utilizaron se basaba en la utilización de un algoritmo de aprendizaje supervisado, conocido como K-means, y otro algoritmo, puramente matemático (Análisis Discriminante de Fisher).

1.4.2. Emotion recognition using physiological signal from multiples subjects²:

Es similar al anterior, pero ellos tomaron un enfoque mucho más centrado en el análisis de las bioseñales. Debido a las limitaciones tecnológicas, el enfoque de este proyecto no sirvió de referencia para nuestro trabajo.

Sus conclusiones demostraron que mediante el estudio del ECG, así como el SKT (Skin Temperature), y el SC (Skin Conductance) son suficientes para obtener con un ratio de acierto elevado una distinción entre las diferentes emociones.

² [Lian Li, Ji-Hua Chen \(2006\)](#)

Además, empezamos a detectar un patrón con respecto a la anterior investigación de la Dra. Nasoz, para la utilización de pequeños videos y cortos para la inducción de emociones.

1.4.3. Emotion recognition using multilayer perceptron and generalized feed forward neural network³:

El estudio del profesor Khanchandani y el Dr. Hussain se basaba en la detección de las emociones utilizando la expresión, tono y fuerza de la voz durante los discursos.

Pese a que su investigación se aleja de nuestra intención, sí nos resultó muy interesante la utilización de redes neuronales para el análisis de patrones de las señales. Este análisis se aproximaría bastante al algoritmo que finalmente emplearíamos en nuestro proyecto. No obstante, la red neuronal utilizada aquí fue una red neuronal multicapa con retroalimentación positiva, en otras palabras, un algoritmo supervisado, mientras que nuestra elección fue un algoritmo no supervisado.

1.4.4. Lifestyle management using wearable computer⁴:

Este documento constataba la utilidad de los dispositivos portátiles para la supervisión del estado físico. En él pudimos comprobar la posibilidad de la utilización de los dispositivos Smartphone para gestionar las señales derivadas de los sensores, así como para gestionar el algoritmo.

³ [K.B. Khanchandani, Moiz A. Hussain \(2009\)](#)

⁴ [Daniel Roggen, Bert Arnrich, Gerhard Tröster \(FALTA AÑO\)](#)

1.4.5. Activity-aware Mental Stress Detection Using Physiological Sensors⁵.

Este extenso estudio nos resultó de extremada utilidad para adquirir un mayor conocimiento sobre el tratamiento de las bio-señales.

Pese a que el estudio estaba orientado a la detección del estrés, muchos de sus análisis y conclusiones, así como metodologías, no pasaron desapercibidos por nosotros. En particular nos interesó mucho la analítica completa que hicieron sobre como estudiar las señales del pulso cardíaco, como limpiar de ruido la señal Galvánica.

También nos interesó la utilización del dispositivo Shimmer para la detección del bio-ritmo cardiaco, dispositivo que finalmente sería utilizado en nuestro estudio, junto con el BodyMedia.

Además, tuvimos ocasión de probar la efectividad algunas de sus metodologías para la clasificación de stress, como la Red Bayesiana de Clasificación o los Árboles de decisión.

Aunque finalmente no utilizásemos ninguno de estos métodos en la experimentación final, nos ayudaron mucho durante nuestras investigaciones iniciales para investigar el comportamiento de las bio-señales humanas.

1.4.6. Emotion Research⁶.

⁵ [Feng-Tso Sun,, Cynthia Kuo, Heng-Tze Cheng1, Senaka Buthpitiya, Patricia Collins, Martin Griss \(FALTA AÑO\)](#)

⁵

⁶ www.emotion-research.net

Aunque no es un documento de investigación en sí misma, este grupo de investigadores posee una gran documentación sobre la investigación, tanto a nivel analítico, como a nivel psicológico, de las emociones. Nos ayudó mucho a comprender mejor cómo funcionan las emociones humanas, como inducir mejor.

Otra de sus características más interesantes fue que habían desarrollado una toolbox para matlab bastante potente para la utilización de redes neuronales. Esta toolbox sirvió de referencia en nuestros estudios iniciales con Matlab para las pruebas de nuestros algoritmos.

1.4.7 Physionet⁷.

No es un estudio en sí mismo, sino una base de datos de diferentes proyectos relacionados con investigación biomédica. Nosotros utilizamos algunas de sus bases de datos para la experimentación y testeo de nuestros algoritmos durante la fase inicial del proyecto. En particular la utilización de la base de datos del estudio Detecting during real-world driving task using physiological sensors (Pickard, 1991) por el parecido de las señales biométricas con respecto a nuestro experimento.

1.4.8 Otros proyectos.

Gracias al artículo Physiological Signals Based Human Emotion Recognition: A review (Jerryta S, M Murugappan, R Nagarajan, Khairunizam Wan, 2011) pudimos encontrar un correcto resumen de otros diferentes proyectos, así como de

⁷ <http://physionet.org/physiobank/database/>

las técnicas utilizadas en los mismos para la detección y el reconocimiento de las señales. Los proyectos fueron los siguientes:

1.5. Visión general del documento, estructura.

El presente documento está estructurado del siguiente modo:

En primer lugar, se ha realizado una primera toma de contacto del proyecto y su contexto en el Capítulo 1, Introducción. Este ha comenzado con una descripción del sistema a modo de introducción, seguida de la motivación que ha llevado al desarrollo de este proyecto. Finalmente se han expuesto los objetivos esperados y los estudios en los que nos basamos.

A continuación, se procede a explicar las diferentes características del sistema, comenzando por un manual de usuario en el Capítulo 2, Guía de Uso, donde se expone la forma de usar la herramienta mediante la interfaz gráfica proporcionada. Esta guía abarca también el proceso necesario para instalar la aplicación móvil y los distintos complementos necesarios para un uso completo del proyecto.

El diseño realizado para el sistema, incluyendo la especificación de los algoritmos usados en la SOM de clasificación de señales, en el tratamiento estadístico de los datos y el diseño de las pruebas de inducción emocional, se presenta en el Capítulo 3, Diseño y Especificación. En él, se detallan los requisitos que poseen las funcionalidades de la aplicación. En el resto del capítulo se expone la especificación de los algoritmos desarrollados y la investigación preliminar realizada.

En el siguiente Capítulo, Arquitectura, se detalla de forma exhaustiva la implementación del sistema mediante el estándar UML. Cada sección de este capítulo hace referencia a cada uno de los módulos en los que se ha dividido la arquitectura del sistema. En ellas, se enseña la estructura diseñada e implementada en seis piezas claves en la realización de este proyecto: Controlador, Captura y

procesado de datos, SOM, Envío y recepción de datos al servidor, Tratamiento de datos en el servidor e Interfaz gráfica.

En el Capítulo 5 se explica y analiza el proceso de experimentación seguido, desde la evaluación de la solución adoptada donde contamos el porqué de ese método experimental hasta el análisis de la eficiencia del mismo. Incluyendo la muestra experimental y los resultado obtenidos.

El contenido principal del documento recae en el Capítulo 5, Conclusiones. En él, se repasa el desarrollo del proyecto y los objetivos cumplidos a lo largo del mismo, dando una visión global del sistema implementado. Posteriormente se procede a explicar los diferentes usos y mercados a los que está destinada la aplicación y las posibles ampliaciones que se podrán realizar sobre el sistema si se continuara su desarrollo.

De forma adicional, el documento presenta dos anexos en los que se detallan las tecnologías usadas en los diferentes módulos que componen la aplicación y un Apéndice con las frases y músicas usadas en el proceso de inducción emocional.

2. Diseño y Especificación

2.1. Introducción

Este capítulo se centrará en el análisis y especificación realizados sobre el sistema, previos a la implementación. Para ello, se detallarán la captura de requisitos que definen la funcionalidad del sistema, para luego analizar con detalle los algoritmos diseñados para llevar a cabo las funciones más importantes de la aplicación.

2.2. Metodología

2.2.1. Flujo completo de información

A continuación mostraremos el flujo completo de los datos desde que son recogidos y pasan por todos los procesos en que consiste el ecosistema creado.

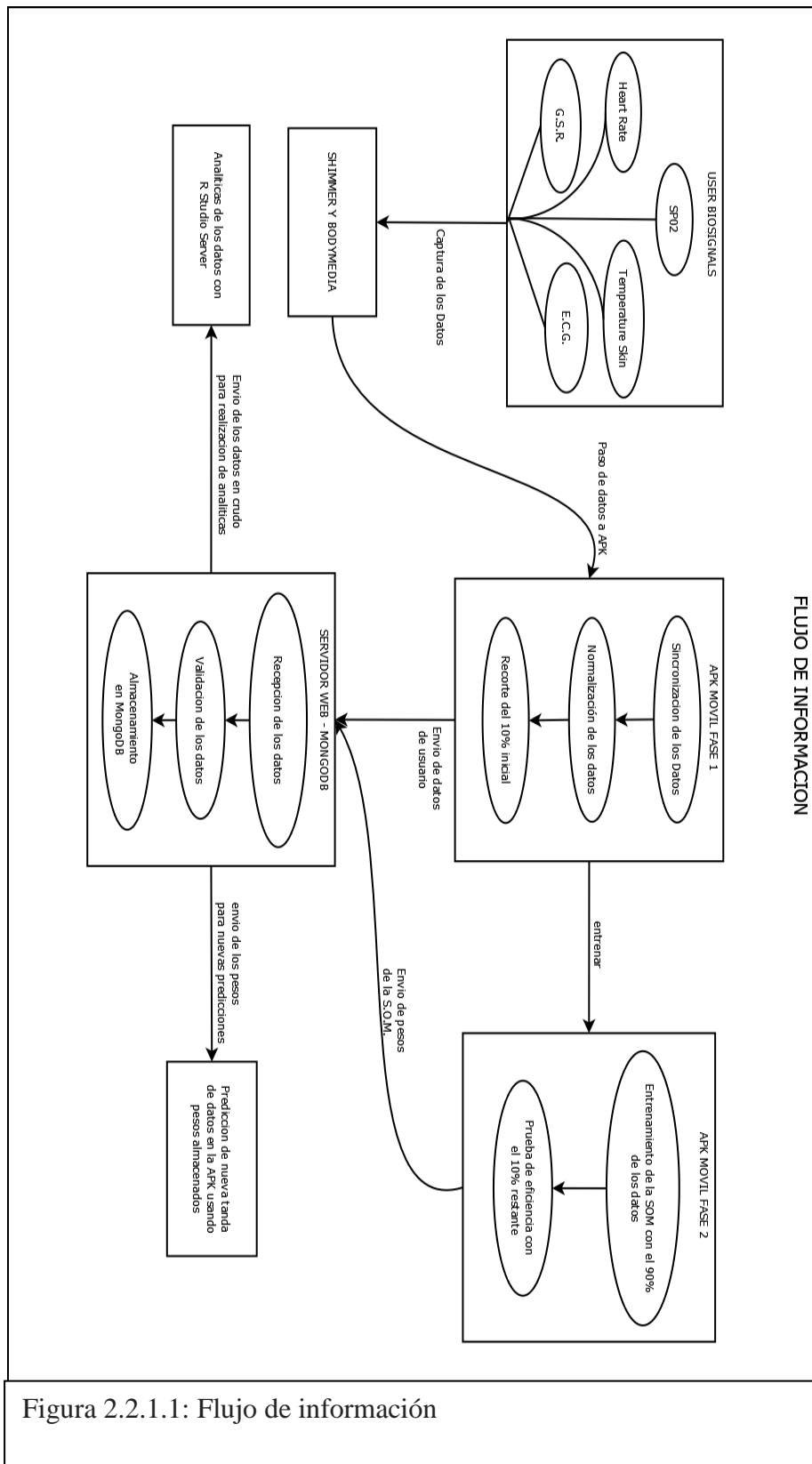


Figura 2.2.1.1: Flujo de información

2.2.2. Hipótesis de partida (Emociones con impacto biométrico)

Nos basamos en que todas las emociones que sufre el cuerpo humano tienen un impacto biométrico diferenciado en menor o mayor escala. Esto se debe a que no tenemos las mismas pulsaciones si estamos asustados a si estamos relajados o a la oxigenación de la sangre, ya que si el cuerpo humano se encuentra en una situación de tensión aumenta la cantidad de oxígeno en la sangre. Algo similar puede decirse de otras variables como la sudoración, la temperatura corporal... las cuales se alteran en el curso de las emociones.

El problema nos surge tras estudiar las señales biométricas, éste es que cada individuo produce unas totalmente distintas a las de otro. Un ejemplo serían las personas que hacen deporte de forma habitual y la gente que no ha hecho deporte nunca o lo realiza de manera excepcional, los primeros tienen un pulso más estable y los segundos no. Igualmente, existen personas que reflejan mayormente las emociones, mientras que otras pueden parecer prácticamente insensibles a éstas. Esto nos impide generalizar un método para todo el 100% del mundo y provoca que debamos crear instancias especializadas en cada individuo, ya que en el caso de no hacerlo la clusterización de las emociones sería imposible ya que dos individuos distintos podrían producir señales biométricas parecidas para distintas emociones y también las producirían totalmente distintas en caso de inducir la misma emoción.

2.2.3. Captura de variables biométricas e inducción de estados anímicos.

La captura de las variables biométricas se realizará a través de dos dispositivos:

1. El primero de ellos se denomina Shimmer (<http://www.shimmer-research.com/>) el cual es un dispositivo basado en el microprocesador MSP430 que transmite los datos recogidos en las interfaces ADC a través de bluetooth.

Este dispositivo debe colocarse en el torso del sujeto mediante cuatro parches tal y como indica el propio Shimmer e insertar en él un dispositivo de fotopleximografía para poder capturar la cantidad de oxígeno en sangre y pegárselo al sujeto en el dedo.

Una vez preparado todo esto se debe reiniciar el Shimmer y asegurarnos de que esta encendido. Después sincronizaremos el dispositivo a la aplicación Green_ECG la cual generará los logs que usaremos después.

2. El segundo de ellos es el brazalete BodyMedia (<http://www.bodymedia.com/>) el cual es un dispositivo que se usa para calcular el consumo calórico y otras variables biométricas del cuerpo.

El modo de uso es muy sencillo necesitando nada más que el usuario se coloque el brazalete en el brazo y espere a que se encienda, lo cual se indica con una señal sonora.

Para generar los ficheros pertinentes tendremos que descargarlos tal y como se ha explicado anteriormente.

Las señales biométricas capturadas son las siguientes:

- ECG: El electrocardiograma es la representación gráfica de la actividad eléctrica del corazón
- SP02: Saturación porcentual de oxígeno en sangre.
- Ritmo cardiaco: Período armónico de latidos cardíacos formado por los sonidos de Korotkoff.
- Temperatura: Indica la temperatura a la cual esta el cuerpo del sujeto.
- Respuesta galvánica de la piel: Mide la resistencia eléctrica de la piel.

Una vez terminado el entrenamiento se descargan los datos y se introducen en la memoria de nuestro dispositivo Android en las carpetas *sdcard/EmotionControl/EmocionPertinente*

En el caso de diagnóstico se deberán introducir en la carpeta *sdcard/EmotionControl/Diagnostico/*

2.2.4. Desarrollo de algoritmos de clusterización de señales

Para realizar la clasificación de los datos en distintas nubes de puntos usamos un *Mapa Auto Organizado de Kohonen* con una topología matricial de 10x10 neuronas y un radio de alcance entre ellas de 5 puntos. Cada una de estas neuronas tiene 5 pesos, indicando cada uno de los parámetros a tener en cuenta: ECG, SP02, ritmo cardiaco, temperatura y la respuesta galvánica de la piel, en este orden y todos ellos estandarizados, de manera que se encuentren en la misma escala. Como inicialización de los pesos de la neurona usamos una inicialización aleatoria (por defecto en la librería usada).

A la hora del aprendizaje, usamos un factor de aprendizaje decreciente de 0.05 a 0.01, una *función de vecindad Gaussiana* y para el cálculo de la distancia entre datos y neuronas, y neuronas con otras neuronas *un sistema métrico euclídeo*. Con estos parámetros hemos encontrado un buen rendimiento y por ello han sido escogidos.

2.2.5. Evaluación experimental con síntesis emocional

Una vez preparado el sujeto ha de prepararse la sala de pruebas, la cual debe poseer un ordenador y la posibilidad de graduar la cantidad de luz que entra en la sala. Se usarán videos, frases y músicas explicadas anteriormente.

En el cuadro siguiente se muestra la luminosidad que debería tener la sala y si ha de tener video o frases y musica.

Emoción	Luminosidad	Video	Música y Frases
Relax	Alta	No	Si

Miedo	Nula	Si	No
Ira	Media	No	Si
Tristeza	Baja	No	Si

El sujeto ha de estar solo en todo momento a lo largo de realizar las pruebas y deberá apuntarse tanto el instante de inicio como el de fin.

Las pruebas durarán en torno a cinco minutos con tres minutos de descanso entre cada una.

En los casos en los que el usuario deba escuchar musica y frases se ha de tener en cuenta que la música excepto en el caso de la Ira ha de ser ambiental y en el caso concreto de la Ira ha de estar a un nivel alto. Hay que explicarle al sujeto que deberá leer unas frases en voz alta e interiorizándolas lo máximo posible como parte de la prueba, se ha de intentar que el usuario se lo tome en serio ya que en caso contrario la prueba se consideraría nula.

2.3. Requisitos

Esta sección describe las funcionalidades del sistema: su comportamiento esperado así como sus restricciones de rendimiento y capacidad. Aunque el proyecto que nos ocupa ha sido desarrollado sin ningún cliente específico, se han elaborado requisitos basados en nuestra visión de la aplicación como conjunto, de forma que tanto la planificación como el desarrollo se pudieran abordar más fácilmente.

A continuación se procede a determinar cuáles son los requisitos funcionales y los no funcionales.

2.3.1. Requisitos funcionales

En el caso de la interfaz gráfica deberá ser capaz de:

1. Guardar y almacenar los pesos de la SOM.
2. Empezar un entrenamiento para cada una de las emociones.
3. Empezar un diagnóstico.
4. Mostrar el resultado del diagnóstico

En el caso del controlador deberá ser capaz de:

1. Coordinar las funcionalidades del resto de módulos de la aplicación.

En el caso del servidor deberá ser capaz de:

1. Recibir y almacenar correctamente la información recolectada durante las fases de entrenamiento de cada uno de los usuarios.
 2. Recibir y almacenar la información de las variables generadas por el mapa autoorganizado para la recuperación del mismo cuando sea necesario.
 3. Ser capaz de entregar la información necesaria, tanto de los datos de entrenamiento, como de los pesos de la S.O.M. siempre que se lo solicite.
 4. Poseer las herramientas necesarias para el análisis remoto de la información recolectada.

En el caso de la SOM deberá ser capaz de:

1. Recolectar todos los datos capturados y generar una red neuronal de pesos afines a ellos de manera no supervisada
2. Clasificar los datos de diagnóstico según los pesos generados en la red neuronal

2.3.2. Requisitos no funcionales

En el caso de la interfaz gráfica deberá ser capaz de:

1. Amable con el usuario
2. Funcional en Android

En el caso del controlador deberá ser capaz de:

1. Poseer una estructura que le permita modificar fácilmente el comportamiento de la aplicación

En el caso del servidor deberá ser capaz de:

1. Realizar complejos cálculos previos que garanticen la calidad de la información enviada a la aplicación.
2. Poseer una herramienta de visualización online que permita comprobar la calidad de los datos así como de los resultados.

En el caso de la SOM deberá ser capaz de:

1. Realizar todos los cálculos necesarios manteniendo unos niveles de memoria compatibles con Android.
2. Mantener los retardos de ejecución al mínimo para no producirse retrasos debido a la menor capacidad de los dispositivos móviles.

2.4. Investigación preliminar

2.4.1 Modos de inducir emociones

En la psicología actual se contemplan diversos modos para producir estados emocionales, a continuación explicaremos en qué consisten los utilizados en este proyecto, y posteriormente realizaremos un breve resumen de algunos otros:

2.4.1.1. Frases de Velten (Velten, 1968)

El método de las frases de Velten consiste en la lectura repetida de un conjunto de frases autodirigidas al sujeto del experimento de contenido emocional. Este método, siendo uno de los más estudiados y usados, se ha demostrado especialmente eficaz para la inducción de estados de tristeza y relajación.

El contenido de las frases de Velten puede dividirse en dos grupos según su contenido: contenido somático, “De vez en cuando me siento tan cansado y triste que preferiría sentarme simplemente a hacer nada”, que tienen como objetivo inducir sensaciones físicas relacionadas con la emoción que se está provocando, y

frases de contenido autoevaluativo, “Hay demasiadas cosas negativas en mi vida”, para inducir pensamientos negativos centrados en la visión de su propio entorno. (*A laboratory task for induction of mood states, Velten, 1968*)

Estudios posteriores lograron aumentar la efectividad del *proceso de Velten* reproduciendo durante el mismo distintas pistas musicales relacionadas generalmente con el estado emocional que se busca conseguir. (*Velten and musical mood induction procedures: a comparison with accessibility of thought associations, F A Albersnagel, 1988*)

2.4.1.2. Inducción a través de piezas musicales

Uno de los primeros estudios de inducción emocional a través de piezas musicales (Sutherland, Newman & Rachman, 1982) permitía a los sujetos escoger entre una variedad de piezas musicales asociadas a cada emoción, sin embargo los estudios posteriores, eliminaron esta posibilidad de elección, usando la misma pieza para todos los sujetos. Martin (1990) destacó que la inducción musical inducía el estado deseado más del 75% de las veces, por otro lado Gerrards-Hesse et al. (1994) Y Westermann et al. (1996) concluyeron que la inducción musical era de los métodos de inducción más eficaces.

Los primeros investigadores en usar música (Clark, 1983), asumían que eran ciertas propiedades de la estructura musical. Este hecho es confirmado por estudios más recientes que así lo afirman, demostrando que hay ciertas propiedades que causan que el oyente perciba expresiones emocionales en la música (Gabrielsson & Lindström, 2001). Casi todos los estudios han sido diseñados con música instrumental, sin embargo, un estudio realizado por Stratton and Zalanowski (1994) demuestra que las letras de las canciones tienen un gran impacto sobre el estado anímico. Para comprobarlo, presentó a distintas personas

la misma canción con y sin letras, y con las letras por separado. Los oyentes que escucharon la versión con letra demostraron un mayor impacto emocional que aquellos que habían sido expuestos a las letras únicamente. Al mismo tiempo, aquellos que escucharon la versión instrumental fueron los menos afectados.

Uno de los mayores inconvenientes de la inducción de emociones a través de la música son las preferencias personales, la misma pieza musical puede afectar de distinta manera a unas personas que a otras, lo cual lo convierte en un tema difícil de investigar. Algunas investigaciones se realizaron pasando un test inicial sobre preferencias musicales a los sujetos del experimento (Terezis, 1993), de manera que las pruebas se realizaban sobre distintas posibilidades musicales. (Emotion Induction Through Music: A Review of the Musical Mood Induction Procedure, Daniel Västfjäll 2002)

2.4.1.3. La utilización de fragmentos de películas

Existe gran cantidad de estudios que demuestran que las películas son capaces de provocar distintas emociones (*Gross & Levenson 1993, Karama et al. 2002*), incluso se puede llegar a afirmar que poseen ciertas ventajas con respecto a otros métodos de inducción emocional, aunque no hay estudios al respecto. Los datos publicados indican que aunque hay cortos capaces de generar activación emocional (*Rottenberg, Gross, Wilhelm Najmi & Gotlib, 2002*), otros pueden no tener ningún efecto.

Uno de sus principales problemas es que al ser una serie de imágenes en movimiento, los fragmentos de películas requieren una gran atención, de hecho, la eficacia puede reducirse si tiene que competir con otros elementos externos. Por ello, para poder ser usado para provocar una respuesta emocional, el sujeto ha de encontrarse en un entorno relativamente aislado. Otro gran problema que presenta, es la personalización de los gustos. Al igual que en el caso de la música, las

escenas que para una persona pueden provocar una respuesta intensa, para otra pueden ser inocuas. (*Emotion Elicitation Using Films*, Jonathan Rottenberg, Rebecca D. Ray, James J. Gross, 2007)

Otros procesos para realizar la inducción emocional que han sido usados comúnmente para realizar experimentos son:

2.4.1.4. El recuerdo autobiográfico

Consistente en inducir al sujeto a recordar experiencias pasadas fuertemente asociadas con un estado emocional concreto (*Brewer, Doughtie y Lubin, 1980*). Este método en muchos casos puede ser producido por frases de Velten concretas sobre experiencias del sujeto en cuestión.

2.4.1.5. La sugestión hipnótica

La técnica empleada para inducir emociones usa la imaginación guiada por sugerencias hipnoticas. Se escogen personas fácilmente hipnotizables y se les pide que se pongan alegres o tristes imaginando o recordando momentos en los que han sido extremadamente felices o tristes. Se ajusta la intensidad emocional a un nivel intenso pero sin llegar a un nivel extremo. La ventaja de la hipnosis es que casi cualquier emoción puede ser producida rápidamente y a una intensidad controlada. (*Mood and Memory*, Bower, 1981).

2.4.1.6. La manipulación de la expresión facial

Se han realizado numerosos estudios sobre los efectos de las expresiones faciales para producir o reforzar los efectos de otros métodos de inducción emocional. Dichos estudios comprobaron que cuando el sujeto se ve obligado a formar expresiones faciales representativas de alguna emoción concreta, sea debido a la inclusión de elementos externos que mantienen la expresión (Martin,

Harlow & Strack 1992) o proporcionando refuerzos positivos durante una entrevista (Kleinke y Walton 1982).

2.4.1.7. Solución adoptada

Finalmente nos decantamos a utilizar el método de las frases de Velten con un refuerzo auditivo para las emociones de tristeza, relax e ira y videos para la emoción de miedo.

Elegimos estos métodos porque son los que consideramos más eficientes a la hora de la inducción de emociones.

2.4.2. Métodos estadísticos estudiados

Durante la fase previa de experimentación se utilizaron, como ya se mencionó previamente, los datos del experimento “Detecting during real-wold driving trask using physiological sensors” (Pickard, 1991). Para su análisis se aplicaron algunos algoritmos recomendados en otros estudios preliminares y pudimos ver su eficiencia. Las fases que componen este análisis fueron las siguientes.

1. Estudio sobre las características de los datos. Estos fueron los datos sobre los que se experimentó:

- Heart Rate:
 - Mínimo = 43.
 - Máximo = 111
 - Media = 75
 - Desviación Típica = 8
- GSR (low):
 - Mínimo = 0
 - Máximo = 13855
 - Media = 6629

- Desviación típica = 1871
- GSR (High):
- Mínimo = 4375
- Máximo = 18337
- Media = 11517
- Desviación típica = 2911

Además de estos datos, también se analizó la simetría y curtosis de los datos, dando como resultado que el pulso sanguíneo tenía una mayor tendencia a la Distribución Normal que las otras dos características.

2. Mediante la utilización de Matlab, se realizó una normalización de los datos, para que todos ellos conservasen el mismo escalado entre 0 y 1, manteniendo la proporcionalidad entre ellos. La fórmula utilizada para los datos fue la siguiente:

$$\frac{x_i - \text{Min}(x_i)}{\text{Max}(x_i) - \text{Min}(x_i)}$$

3. Reducción del número de datos. Debido a la cantidad de datos por cada una de las columnas (82442 filas), fue necesario implementar una reducción del número de los mismo para poder implementar algoritmos complejos sin que Matlab obtuviera errores debidos a falta de memoria. Por ello, se aplicó un algoritmo de reducción consistente en obtener el valor medio de 10 valores consecutivos y utilizarlo como dato final. De esta manera se redujo el tamaño del experimento a un décimo de su tamaño original. Cantidad suficiente para que Matlab pudiera trabajarla sin problemas de memoria.

4. Eliminación de outliers: Al estar trabajando con sensores biométricos, existe una alta probabilidad de encontrar datos erróneos debidos a una falta de contacto del sensor o a que el mismo se mueve de su posición, etc. Por ello, se

realizó un análisis de valores extremos y outliers aplicando el algoritmo MEDA . Este algoritmo se rige mediante la creación de un cubo basado en los límites generados por la diferencia entre el valor medio de la columna de datos y su valor MEDA. Este valor se obtiene realizando la mediana del valor absoluto de la diferencia entre el valor y su media.

5. Aplicamos un algoritmo k-Means para obtener una primera aproximación de los diferentes grupos de emociones (en este caso intentamos dividir los datos en 3 grupos, relax, estrés leve, estrés alto).

6. Una vez obtenidos los grupos, se realiza un muestreo del 10% de los datos para entrenar los diferentes algoritmos que se probarán intentando mantener un porcentaje equilibrado de datos en cada uno de los grupos.

7. Con los datos de entrenamiento se entrena un algoritmo de Análisis Discriminante Canónico de Fisher. Este algoritmo pretende describir tanto algebraicamente como gráficamente las diferentes características de las observaciones que provienen de varias poblaciones. Se trata de buscar variables discriminantes (reales o ficticias) cuyos valores propicien la separación de una población en subpoblaciones específicas tanto como sea posible.

8. Posteriormente se prueba el algoritmo que utilizaremos, la red neuronal de Kohonen, o mapa autoorganizado, se entrena con la misma muestra que se entrenó el Análisis Discriminante.

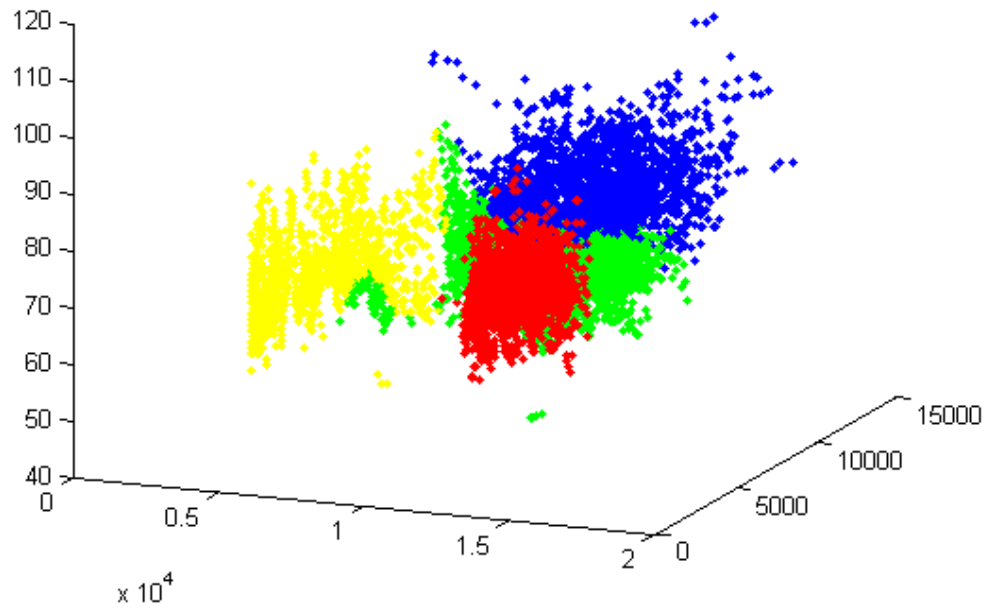


Figura 2.4.2.1: Clasificación de los datos obtenida de estrés de conductores

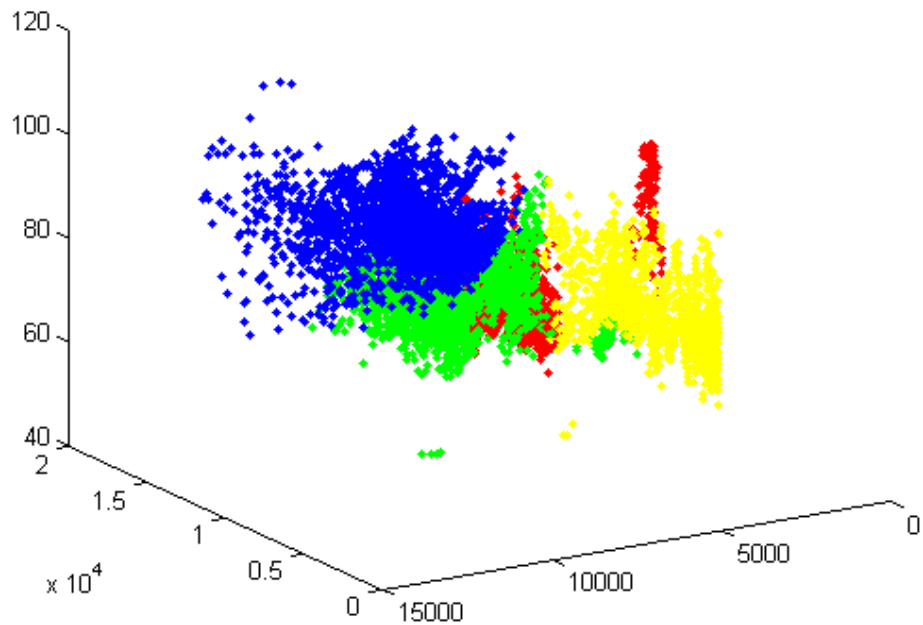


Figura 2.4.2.2: Clasificación de los datos obtenida de estrés de conductores, visto desde otro ángulo

2.4.3. Distintos algoritmos de clasificación, librerías

Para la investigación preliminar de algoritmos de clasificación, realizamos pruebas sobre distintos algoritmos de clasificación automática. De entre todos ellos, escogimos las redes neuronales por su gran versatilidad y entre ellas, los *Mapas Autoorganizados de Kohonen*, por su gran versatilidad y la ventaja de no necesitar supervisión tras el entrenamiento inicial.

Una vez decidido el tipo de algoritmo que íbamos a usar, nos centramos en el entorno Matlab como entorno para realizar las pruebas. Tras una búsqueda de las librerías sobre redes neuronales más completas y de un periodo de prueba con ellas, nos decantamos por la utilización de la librería SOM Toolbox (<http://www.cis.hut.fi/somtoolbox/>). Usando esta toolbox de Matlab realizamos las primeras pruebas comprobando los distintos parámetros que ofrecían los *Mapas Autoorganizados de Kohonen* para decidir los más eficientes, además de familiarizarnos con su uso y funcionamiento.

Para realizar el paso del entorno de trabajo a java, entorno usado por Android, experimentamos con diversas librerías de SOM. Inicialmente, escogimos la librería *Java Kohonen Neural Network Library* creada por Janusz Rybarski y Seweryn Habdank-Wojewódzki de licencia *Nueva BSD*. Esta librería contiene toda la funcionalidad requerida para implementar un *Mapa Auto Organizado de Kohonen*. Pero debido a problemas de implementación que presentaba, usamos una adaptación del paquete *Kohonen* de código abierto para R, publicado por Ron Wehrens, para poder aplicarla desde el sistema operativo Android.

2.4.4. ¿Dónde realizar el proyecto? SOs, Eclipse, etc

El proyecto se desarrolló en varios sistemas diferenciados debido a las características específicas que tenían cada una de las partes:

- Aplicación Móvil: La aplicación móvil se desarrolló sobre un entorno virtual de linux, distribución Ubuntu 12.04 LTS. Se desarrollo bajo JAVA, utilizando el IDE Eclipse para ello, con el conector Android-Java necesario.

- Servidor (implementación): El servidor se desarrolló sobre un entorno virtual Linux, distribución Debian, se desarrolló bajo Java, Maven y Spring, utilizando el IDE NetBeans para su desarrollo, por las facilidades para administrar los repositorios de Maven. De igual forma, R y MongoDB están preconfigurados en el pc para las pruebas.

- Servidor (montaje): El servidor esta montado sobre una distribución de Ubuntu Server 12.04 en la nube de Amazon AWS, utilizando Rstudio Server, se puede acceder online para la realización de las analíticas.

2.4.5. Algoritmos usados

2.4.5.1. Estadísticos

Para el análisis de los resultados se realizó una analítica de los datos recogidos basado en los siguientes estadísticos.

- Contraste de bondad de ajuste: Una de las primeras comprobaciones fue asegurar si los datos de cada uno de los sensores se ajustaban a una distribución Normal. Este análisis consiste en la realización de un test Chi-cuadrado donde la Hipótesis nula es la igualdad de los valores esperados en una normal estándar, con media y desviación típica iguales a los de la distribución, y los valores reales.

- Análisis de correlaciones: Antes de evaluar los distintos algoritmos de clasificación, evaluaremos un análisis de correlaciones de las variables, este análisis nos dirá, entre otras cosas, la relación que guardan entre sí (de manera lineal) las variables implicadas en nuestro análisis. En particular nos interesa saber

la relación que guarda nuestra variable dependiente emoción, con las variables de los sensores. Así como comprobar que no existe demasiada relación lineal entre las variables predictoras (cuando más independientes sean entre sí, mayor será la tasa de acierto). Se realizará una matriz de correlaciones para cada señal. Recordemos que la fórmula de la razón de correlación para una variable es:

$$\frac{\text{COV}(x, y)}{\sigma(x) * \sigma(y)}$$

- Análisis de la homogeneidad de las Varianzas: Para poder realizar algunos de los algoritmos, se ha de asumir que se cumple homogeneidad de varianzas-covarianzas entre las variables. Por ello aplicaremos el test de homogeneidad de Bartlett y el test de homogeneidad de Figner-Killeen (uno para caso paramétrico y otro para el caso no paramétrico). La representación se hará mediante el Análisis Brown-Forsyth de varianzas homogéneas

- Análisis Discriminante: Dependiendo de si hemos encontrado homogeneidad en las variables o no, se realizará un análisis discriminante lineal, o un análisis discriminante cuadrático. Este análisis será la primera prueba que nos dirá las agrupaciones teóricas. Para su realización, se entrenará con una muestra aleatoria idénticamente distribuida del 60% de los datos y se comprobará con un 40% de los mismos.

2.4.5.2. Inteligencia artificial

Los *Mapas Autoorganizados de Kohonen* son un modelo de red neuronal no supervisada usada para la clasificación de datos en distintos grupos según su cercanía entre sí. Los SOM (del inglés *Self Organizing Map*) se componen de dos

capas de neuronas, la capa de neuronas de entrada, encargada de recibir y transmitir a la capa de salida la información procedente del exterior. Y la capa de salida, encargada de procesar la información y formar el mapa de grupos. De esta forma, cada neurona de salida tiene asignado un vector de pesos que será modificado según se introduzcan los datos de entrenamiento.

A la hora del entrenamiento, se utiliza aprendizaje competitivo, se realizará un cálculo de la distancia entre el vector de datos introducido y todos los pesos de las neuronas. Una vez detectada la neurona más próxima, con los datos más similares, se aplica una fórmula de aprendizaje. En el caso de esta aplicación es usada la fórmula *Winner Takes Most*:

$$w(k+1) = w(k) + n * N(i,j) * (x-w)$$

Donde $w(k+1)$ es el vector de pesos de la neurona en la iteración $k+1$, $w(k)$ el vector en la iteración k , n el valor de aprendizaje, $N(i,j)$ el valor de vecindad de la neurona i con respecto a la neurona j , y $(x-w)$ el vector diferencia entre los datos de aprendizaje y el vector de pesos de la neurona. En esta misma fórmula el proceso se repite para cada una de las neuronas de la capa de salida del SOM, modificando los pesos de todas ellas en menor grado que la neurona con los datos más similares.

Otra de las fórmulas más utilizadas es *Winner Takes All*, donde únicamente se modifican los pesos de la neurona más similar:

$$w(k+1) = w(k) + n * (x-w)$$

Este proceso se repite para cada uno de los vectores de entrenamiento y usualmente, si la cantidad de información de entrada es pequeña, tiene un factor de repetición. De forma que se repite el proceso de cálculo N veces para cada vector.

A la hora de realizar la clasificación de los datos de diagnóstico, se aplica el método k -NN sobre los pesos de todas las neuronas de salida. Este método, realiza un recorrido de la lista de vectores de peso de las neurona, calculando las k más

similares al vector de datos a diagnosticar y guardando el valor de clasificación de todas ellas. Una vez encontradas, realiza un cálculo del valor más popular entre los valores de clasificación y clasifica el dato de entrada con respecto a este valor.

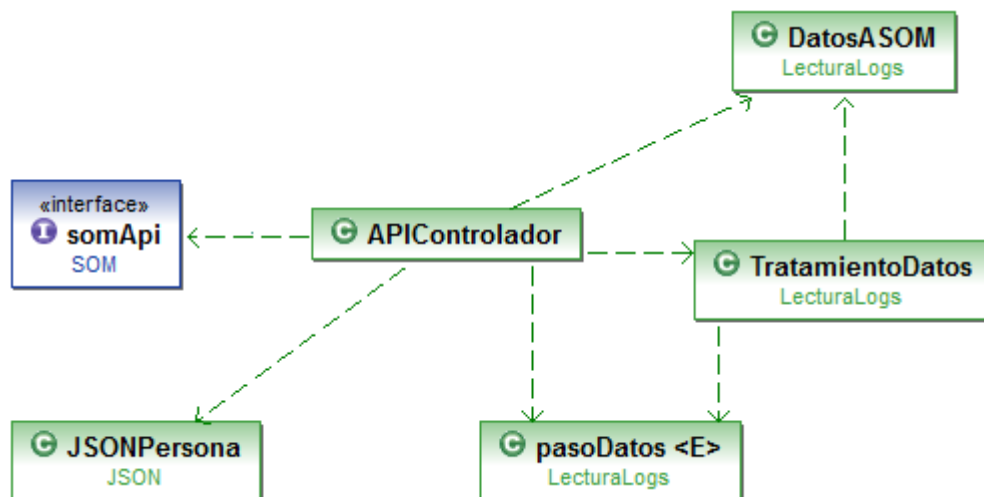
3. Arquitectura

3.1. Introducción

Este capítulo explicará de forma detallada la estructura del sistema desarrollado usando el estándar UML. Este sistema ha sido implementado usando el lenguaje Java.

Se comenzará explicando el tratamiento de datos y su comunicación con el servidor, seguiremos con la SOM y terminaremos con el controlador y la interfaz gráfica.

3.2. Controlador



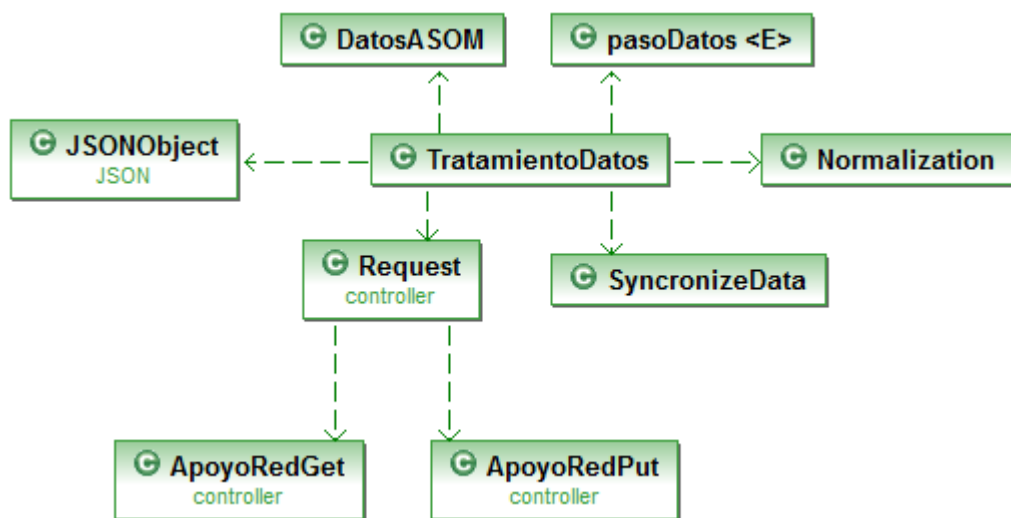
La clase principal es “APIControlador” ya que es la que aúna las funciones principales de la aplicación:

- Comunicación con el Servidor mediante las clases Reques, ApoyoRedGet y ApoyoRed

- Indica que ha de hacer cada momento la SOM mediante su interfaz somAPI
- Gestiona el almacenamiento de datos (JSONObject, JSONPersona) y su posterior tratamiento mediante las clases de pasoDatos, TratamientoDatos y DatosASOM

3.3. Captura y procesado de datos

Los datos son generados mediante el shimmer y el brazalete bodymedia fit. Los datos del primero son enviados mediante bluetooth al dispositivo los cuales los almacenará en logs. En cambio los del brazalete se consiguen mediante un dumpeo de su memoria a csv.



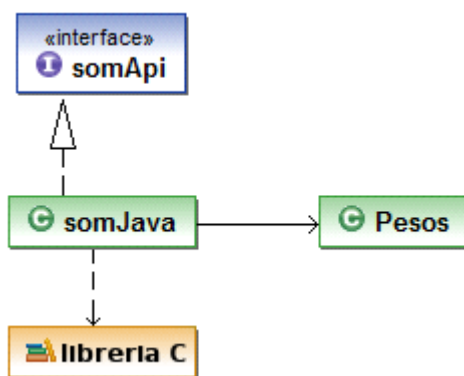
De cargar los datos desde las distintas fuentes se encarga la clase TratamientoDatos. Esta clase se encarga de leer los archivos de datos, usando la clase pasoDatos con el parámetro tipo de los datos, en nuestro caso *double*.

Debido a que en los datos existe una gran disparidad con respecto al momento en el que son tomados, así como de la cantidad de datos recogidos por sensor, es necesario, para poder crear matrices cuadradas que todos los datos tengan la misma longitud y empiecen y acaben al mismo tiempo. De esto se encarga

SynchronizeData, seleccionando el sensor que más datos ha generado y creando mediante una fórmula de valores medios los datos intermedios del resto de sensores. Tras igualar el tamaño de los datos desde las distintas fuentes, es necesario normalizarlos, trabajo realizado por la clase Normalization.

Una vez tratados, se genera un JSON para enviar los datos al servidor de respaldo mediante la clase Request, que genera una petición de envío de datos tramitada por ApoyoRedPut. Por último, los datos son enviados a la SOM mediante la clase DatosASOM.

3.4. SOM



A la hora de describir el funcionamiento de la *SOM*, primero nos encontramos con la clase *somJava* clase fachada que se encarga de la comunicación entre el controlador y las clases de la *SOM*. Esta clase se encarga de generarla, entrenarla, realizar el algoritmo de agrupación y demás funcionalidad relacionada.

Una vez dentro del paquete, nos encontramos la clase *pesos*, usada para mantener actualizada la copia de seguridad del servidor. Y la llamada a la librería que realiza el procesamiento de los pesos neuronales.

3.5. Envío y recepción de datos al servidor

La comunicación con el servidor se hace mediante “HTTP Methods” y se utilizan sobretodo los métodos de POST y PUT para poder así almacenar los datos en la nube y poder tener una copia de seguridad.

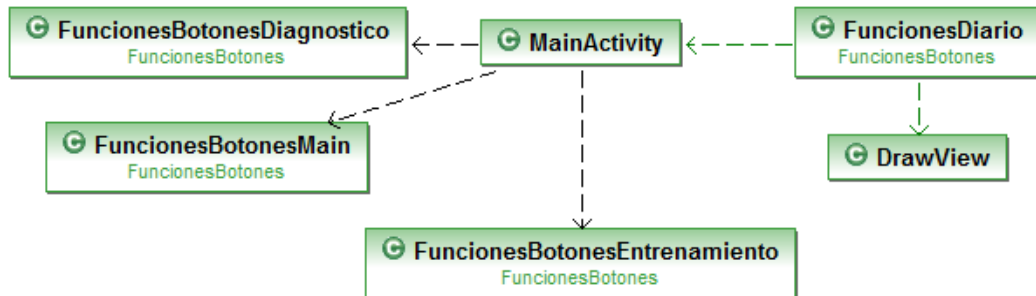
Para la incorporación de estos métodos ha sido necesario implementar la clase Request junto a otras. Cabe destacar la necesidad de la creación de hebras para esta parte, ya que la comunicación con un servidor es siempre asíncrona.

3.6. Tratamiento de datos en el servidor

Los datos son recibidos mediante un servicio REST que recibe y parsea la información en formato JSON según la información de la clase User, o la clase Pesos, si estamos hablando de la información sobre los pesos de la SOM. Posteriormente, la almacena en un servidor de MongoDB para su posterior uso.

El servidor puede dar servicio a peticionarios online mediante órdenes GET, enviando los datos identificados por el id del usuario. También permite, para los datos de entrenamiento, utilizar órdenes PUT para actualizar la información que pudiese haber. Además, posee un servidor de RStudio con la librería RMongo para la realización de pruebas y evaluaciones de los datos.

3.7. Interfaz gráfica



La clase principal de la interfaz gráfica es el MainActivity el cual se encarga de la sincronizar la carga de ventanas y sus respectivos elementos. Para modular la funcionalidad de la interfaz se generaron cuatro clases (FuncionesBotonesDiagnostico, FuncionesBotonesMain, Funciones BotonesEntrenamiento y FuncionesDiario) que contienen la funcionalidad de cada uno de los botones que se muestran diferenciados según su ventana. Esto nos proporciona facilidad a la hora de hacer cambios en la interfaz.

El cambio de nomenclatura en FuncionesDiario se debe a que no posee botones, lo único que contiene es un panel donde se muestra el resultado del diagnóstico. Este panel está contenido en la clase DrawView.

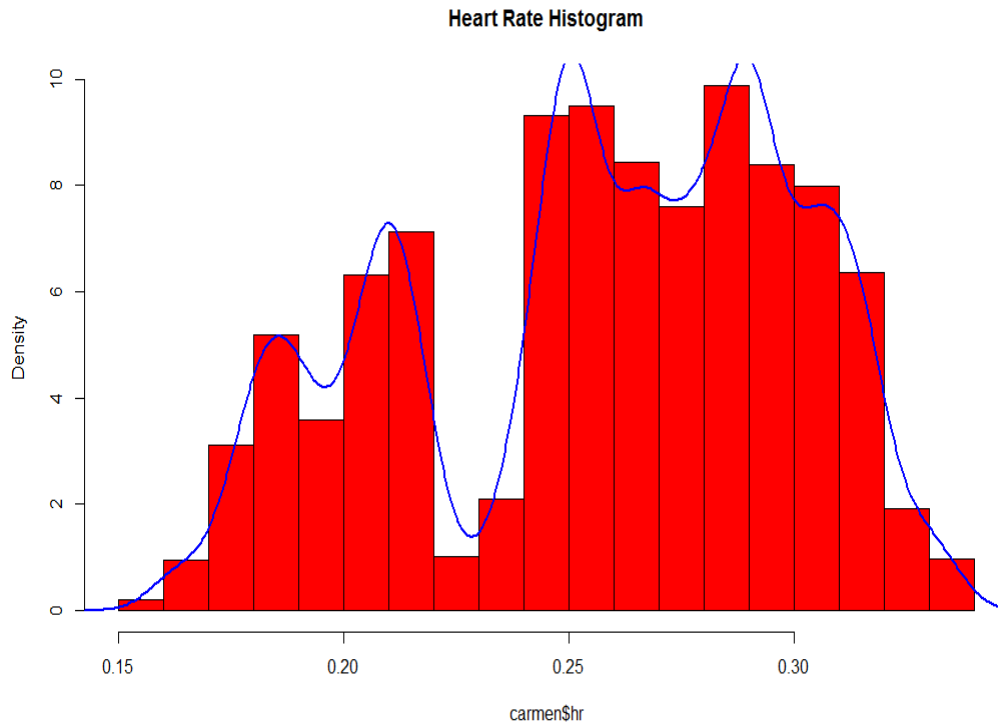
4. PROCESO DE EXPERIMENTACIÓN

4.1. EVALUACIÓN EXPERIMENTAL DE LA SOLUCIÓN ADOPTADA

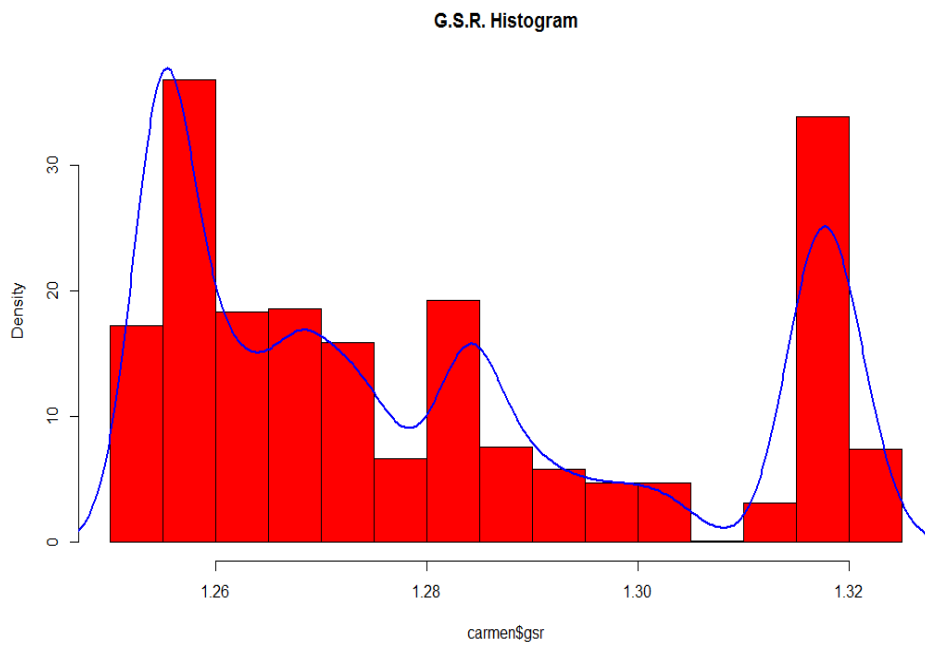
Tal y como se explicó en la sección Estadísticos utilizados, se va a proceder al análisis de los resultados. Para ellos se analizarán inicialmente los datos del usuario Carmen, y posteriormente se extenderá a un resumen general para los demás usuarios:

Las variables de las que se disponen son las siguientes, con su histograma y su función de densidad. Los datos han sido normalizados, por lo que en estos análisis no se mostrarán los valores reales, además, estos son los datos con las 4 emociones tenidas en cuenta:

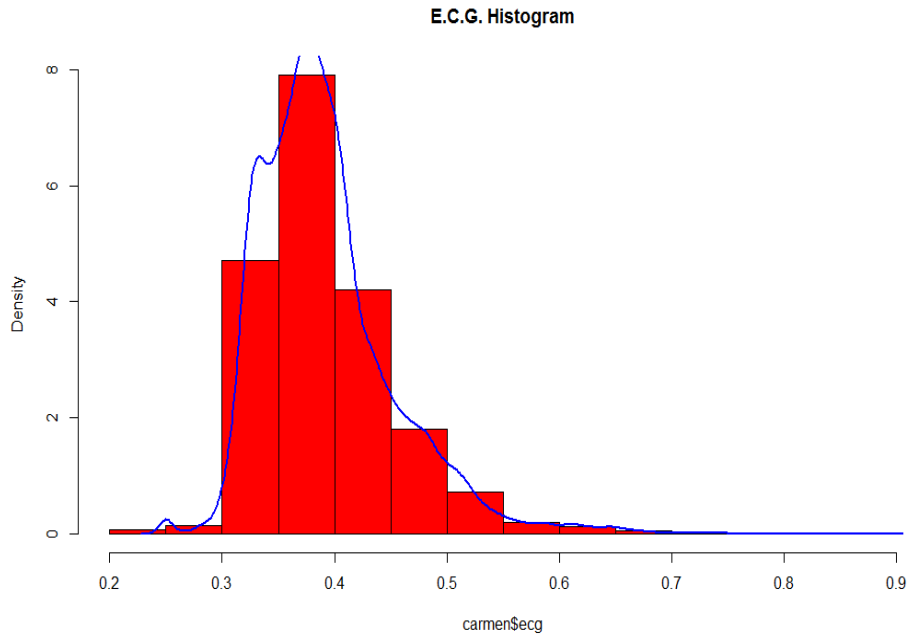
○ HeartRate:



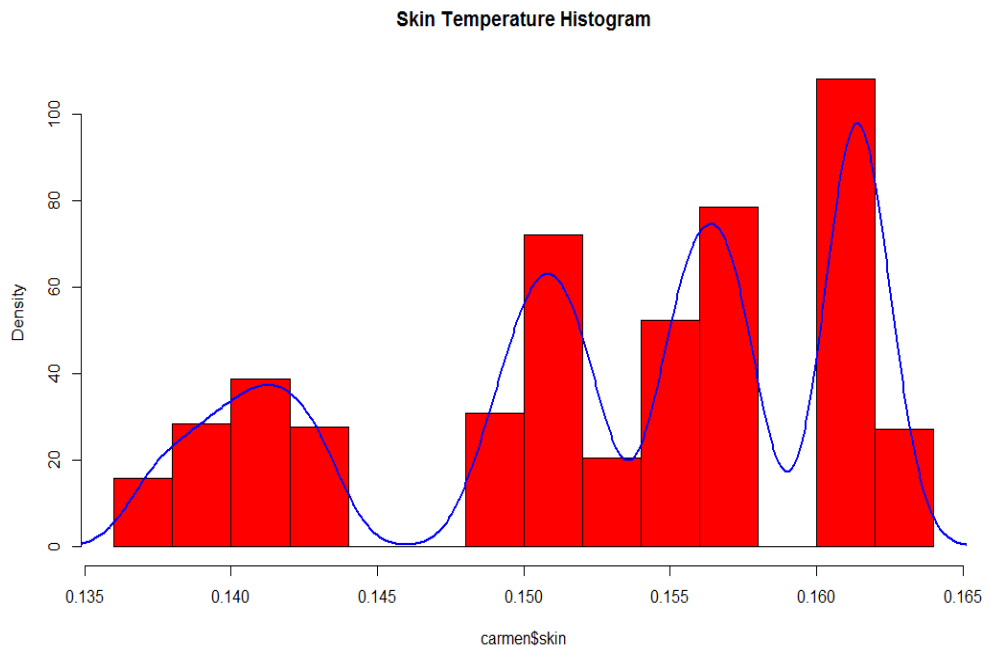
○ G.S.R.:



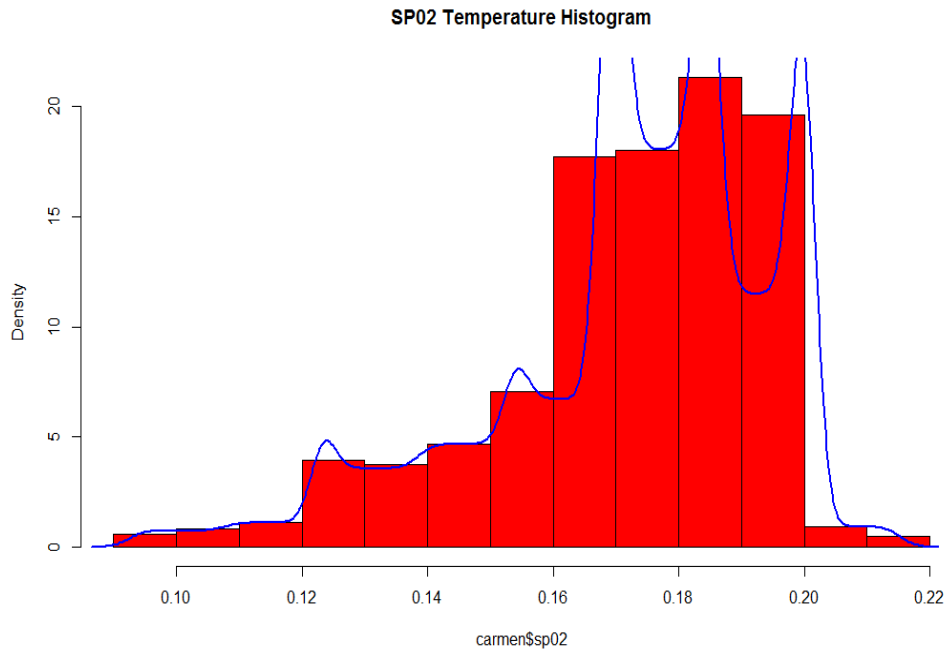
- E.C.G



- Skin Temp

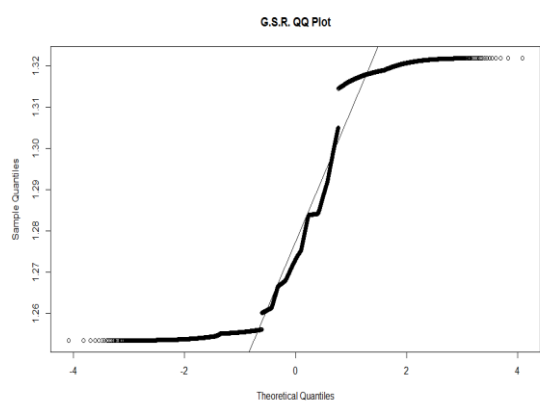
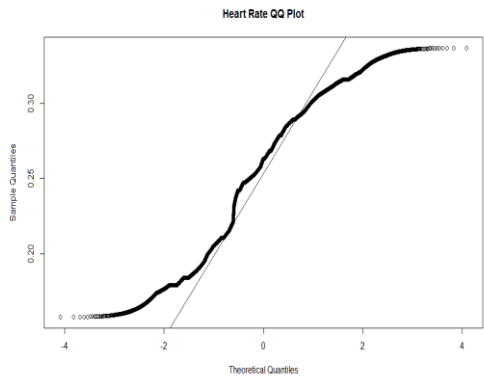


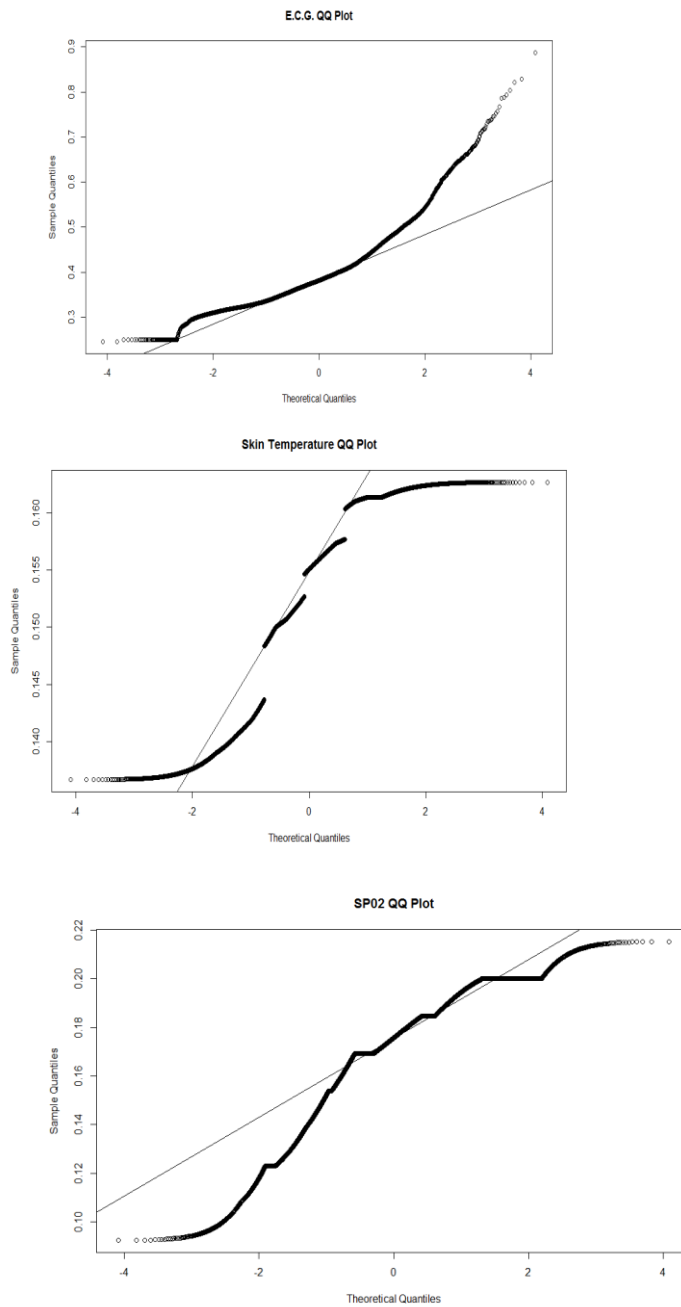
- SP02:



Como puede comprobarse, la mayoría de los datos no poseen acercamiento a la distribución Normal con media 0. Destaca el caso de ECG que posee varios valores outliers, al ver su cola muy extendida hacia la derecha. También el caso de HeartRate posee un mínimo local justo localizado en la media.

Una vez vistos los histogramas con las densidades de nuestras variables, el siguiente paso es comprobar las hipótesis de normalidad:





Como podemos ver en los gráficos de residuos, ninguna de las variables podría considerarse Normal, dado que sus residuos se alejan bastante de la distribución teórica de residuos normal.

Analicemos la homogeneidad de las varianzas en relación a los grupos, como casi todos los grupos tienen alejamiento de la normal, usaremos el test de Fligner-Killeen, pues es menos sensible a desviaciones de la normal.:

- HeartRate::

Fligner-Killeen test of homogeneity of variances

data: hr by emotion

Fligner-Killeen:med chi-squared = 581.4913, df = 3, p-value < 2.2e-16

El p-valor obtenido nos obliga a rechazar la Hipótesis de igualdad de varianzas.

- G.S.R.:

Fligner-Killeen test of homogeneity of variances

data: gsr by emotion

Fligner-Killeen:med chi-squared = 6372.952, df = 3, p-value < 2.2e-16

- E.C.G.:

Fligner-Killeen test of homogeneity of variances

data: ecg by emotion

Fligner-Killeen:med chi-squared = 1996.891, df = 3, p-value < 2.2e-16

- Skin Temperature:

Fligner-Killeen test of homogeneity of variances

data: skin by emotion

Fligner-Killeen:med chi-squared = 6451.894, df = 3, p-value < 2.2e-16

- SP02:

Fligner-Killeen test of homogeneity of variances

data: sp02 by emotion

Fligner-Killeen:med chi-squared = 3493.069, df = 3, p-value < 2.2e-16

En resumen comprobamos que ninguna de nuestras variables cumple condiciones de homocedasticidad. Así que aplicaremos distribución cuadrática para clasificarlas.

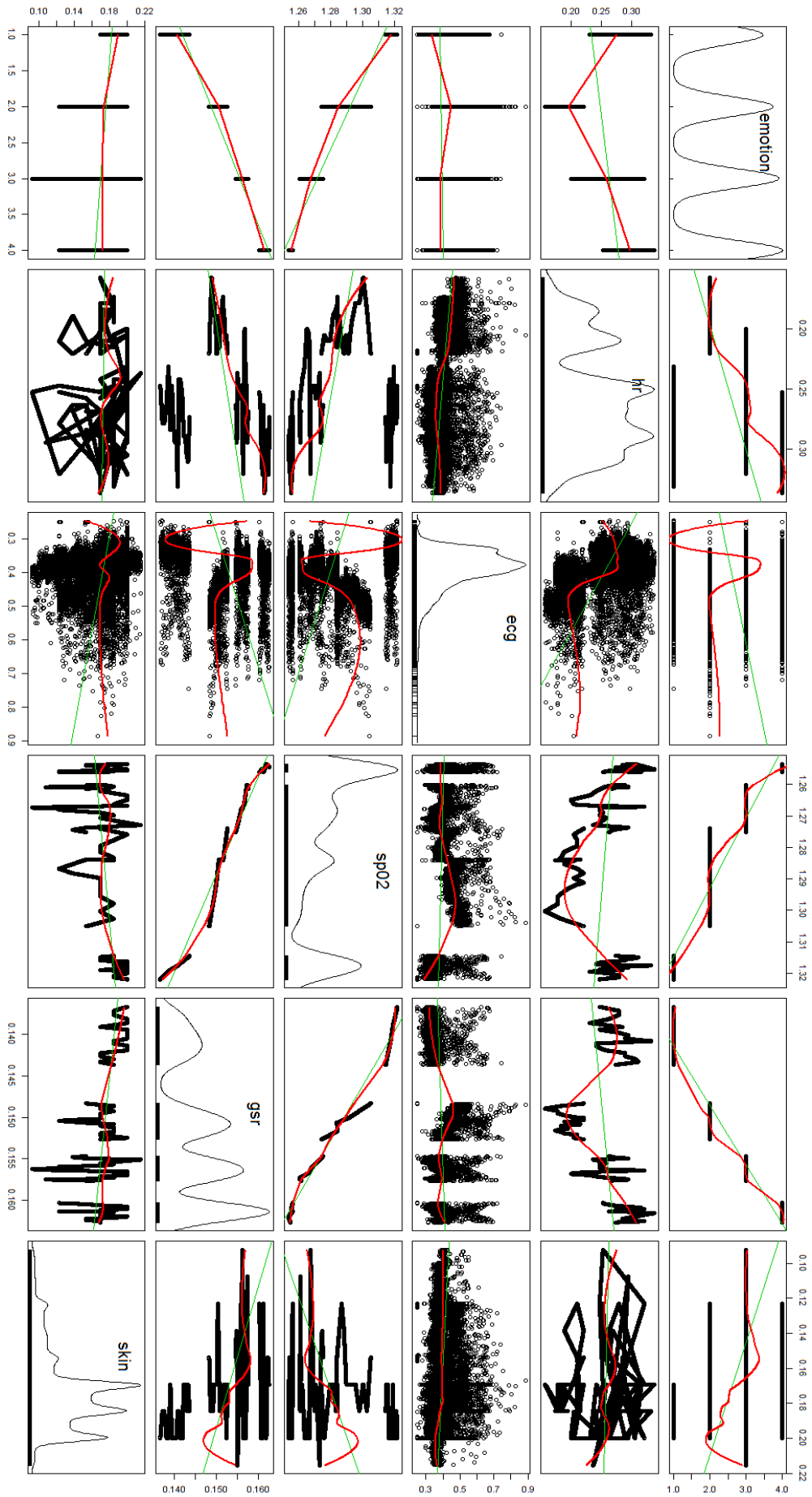
Vamos a analizar la matriz de correlaciones de las variables:

	emotion	hr	ecg	sp02	gsr	skin
emotion	1.0000	0.36945	0.1056	-0.31028	-0.9616	0.9723
hr	0.3695	1.00000	-0.4363	-0.03283	-0.2416	0.2433
ecg	0.1056	-0.43634	1.0000	-0.19000	-0.1699	0.1984
sp02	-0.3103	-0.03283	-0.1900	1.00000	0.3510	-0.3524
gsr	-0.9616	-0.24163	-0.1699	0.35100	1.0000	-0.9896
skin	0.9723	0.24326	0.1984	-0.35236	-0.9896	1.0000

Como podemos observar, la correlación entre la variable regresora y las predictoras es casi 1 en el caso de gsr y skin en todos los casos. Esto implica que estas dos variables han reaccionado muy diferente en cada una de las pruebas, lo que nos ayudará considerablemente durante la detección de la emoción a posteriori.

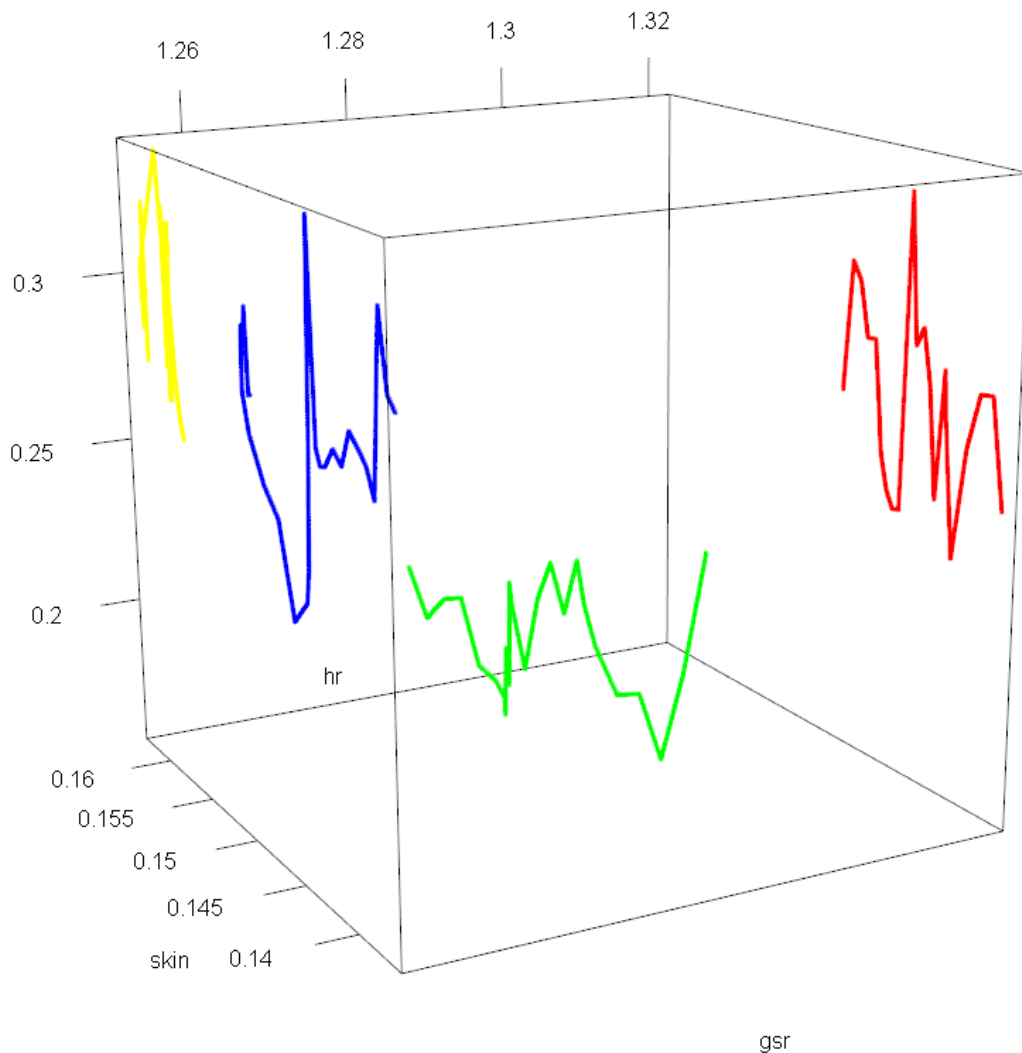
Como detalle adicional, vemos que parece haber cierto nivel de relación GSR y Skin Temperature.

Por último, vamos a representar un gráfico que muestra esta misma tabla de correlaciones de manera más visual. En el veremos algunas de las conclusiones que acabamos de comentar.



El gráfico muestra los puntos entre las variables dos a dos, en la diagonal principal podemos encontrar las funciones de densidad anteriormente mostradas, más una función rug que muestra la concentración de los puntos. En el resto de celdas muestran los gráficos de puntos de las variables dos a dos, un gráfico de regresión lineal y una aproximación de la regresión no lineal que aproxima los puntos.

Sigue destacando la separación marcada entre skin y emotion así como entre gsr y emotion, vamos a representar un gráfico de 3 dimensiones entre las 3 variables más correladas (grs, skin y heart Rate) coloreadas por emoción variables para comprobar mejor hasta qué punto son disjuntas.



Como se puede comprobar, sólo con estas tres variables la separación es casi perfecta, lo cual nos puede dar un indicio de los resultados que obtendremos con nuestras pruebas.

Vistos todos los resultados de las analíticas anteriores, vamos a aplicar un Analisis Discriminante Cuadrático para intentar predecir la información. Seleccionaremos una muestra aleatoria idénticamente distribuida del 70% de los datos para poder entrenar la muestra. Veamos los datos que componen el análisis:

```
distrib$call
qda(formula = emotion ~ ., data = carmen, subset = sampleCarmen)
```

```
distrib$N
```

```
[1] 15836
```

```
distrib$prior
```

```
1 2 3 4
0.2214 0.2499 0.2593 0.2693
```

```
distrib$means
```

```
hr ecg sp02 gsr skin
1 0.2748 0.3407 0.1891 1.318 0.1405
2 0.1967 0.4494 0.1704 1.287 0.1507
3 0.2573 0.3862 0.1639 1.267 0.1563
4 0.2964 0.3867 0.1694 1.255 0.1614
```

```
distrib$counts
```

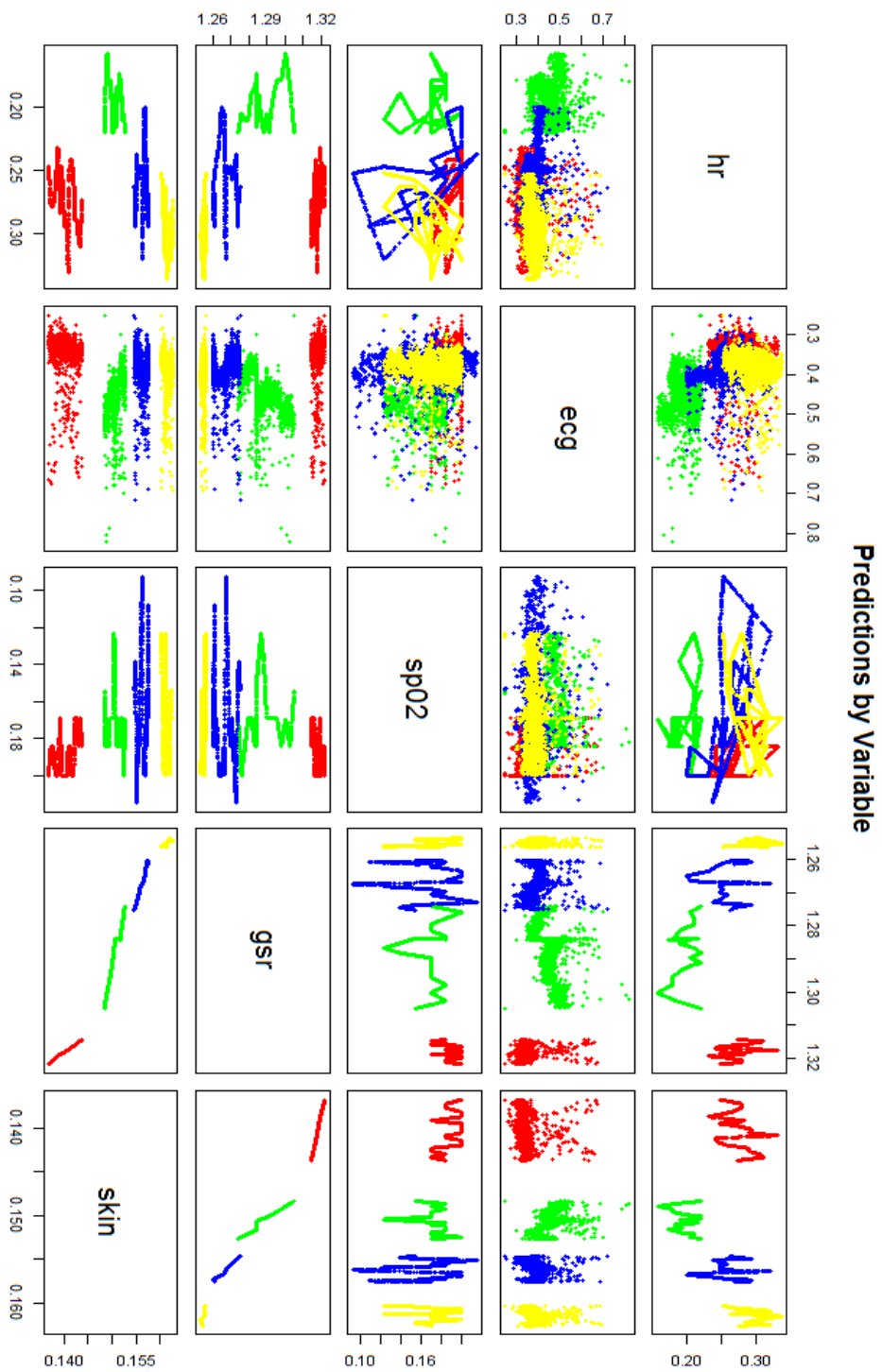
```
1 2 3 4
3506 3958 4107 4265
```

Siendo 1: Alegre, 2: Asustado, 3: Enfadado y 4: Triste

Una vez preparada y entrenada el discriminante, procedemos a realizar una predicción con el 30% restante de los datos y comprobamos y comparamos resultados.

```
1 2 3 4
1 1508 0 0 0
2 0 1672 0 0
3 0 0 1710 0
4 0 0 0 1885
```

En cómputo general vemos que se ha acertado el 100% para cada emoción.



Aquí podemos ver como se han distribuido los grupos en las predicciones para cada una de las parejas de variables, la leyenda de colores es la siguiente: Rojo = alegre, verde= asustado, azul: enfadado, amarillo=triste.

Al realizar las mismas pruebas sobre otros usuarios se obtienen resultados similares, no obstante, si se intenta predecir el comportamiento de un usuario con el entrenamiento de otro, la predicción falla estrepitosamente. No obstante estos análisis los veremos en la sección de verificación de la especificad.

Conclusión: el análisis discriminante cuadrático evalúa las relaciones entre los grupos y las variables regresoras a la perfección en los usuarios entrenando el algoritmo específicamente a cada usuario.

4.2. DESCRIPCIÓN DE LA MUESTRA EXPERIMENTAL (SUJETOS)

La muestra se recogió a partir de compañeros de clase y familiares de los desarrolladores del proyecto. Siendo la muestra original de catorce sujetos pero viéndose reducida a cinco debido a fallos en la experimentación.

El 63.63% de la muestra es de sexo femenino siendo la mayoría personas entre 23 y 25 años, exactamente la media es 27.72 con una desviación típica de 11.7 esto se debe a que hay dos personas de más de cincuenta años y una niña de quince.

4.3. PROCESO DE EXPERIMENTACIÓN Y RESULTADOS OBTENIDOS

Para realizar la verificación de los resultados de la aplicación y tras haber obtenido datos de los distintos sujetos de las pruebas, se han realizado pruebas Leave-one-out de validación cruzada. Estas pruebas consisten en tomar un porcentaje de los datos de entrenamiento de manera aleatoria y no determinista, en nuestro caso un 20% de los datos, e introducirlos como datos de diagnóstico, eliminándolos de los datos de entrada.

Para los datos del sujeto Carmen, se han alcanzado un porcentaje de acierto del 100% para las emociones de Alegría y Miedo, y de un 99% para las emociones de Ira y Tristeza. Se ha escogido este sujeto como muestra de los resultados por ser la menor de las personas de los que se han tomado datos, siendo la que presenta la respuesta emocional más clara.

Para una mayor claridad, se puede observar la distribución de los datos con respecto de las neuronas en el siguiente gráfico:



Figura 5.3.1: Distribución de los datos de Carmen con respecto de las neuronas

4.4. ANÁLISIS DE LA ESPECIFICIDAD Y EFICACIA DE LA SOLUCIÓN ADOPTADA

En la primera sección de este capítulo encontramos que si entrenamos cada usuario, para sí mismo la predicción era perfecta, o rondaba la perfección. No obstante esto no sucedía al utilizar el entrenamiento de un usuario para predecir a otro. Por ejemplo, en el siguiente caso, entrenamos los datos del usuario Samuel, e intentamos predecir los datos de Carmen. Este fue el resultado obtenido:

	1	2	3	4
1	0	5013	0	0
2	0	5588	0	0
3	0	5908	0	0
4	0	6114	0	0

Porcentajes:

1	2	3	4
0	1	0	0

Como vemos, el algoritmo situó todos los datos en la emoción asustado, por lo que se puede decir que no funcionó en absoluto.

No obstante, en la mayoría de ocasiones en las que se entrena un algoritmo y este se pretende que sea genérico se utiliza un conjunto más grande de usuarios unidos para realizar la predicción, en lugar de utilizar solo uno, eso mismo es lo que intentaremos en esta sección. Para ello, se han recogido los datos de 5 personas y se han ido uniendo poco a poco en un mismo conjunto. Veamos los resultados del análisis discriminante, como antes los datos han sido entrenados con el 70% de la muestra y predichos con un 30% aleatorias idénticamente distribuido.

Dos usuarios unidos:

1	2	3	4
0.9985	0.9987	0.9951	0.9985

Tres usuarios unidos:

1	2	3	4
1.0000	0.9810	0.8153	0.9019

Cuatro usuarios unidos:

1	2	3	4
0.9595	0.8779	0.6148	0.6411

Cinco usuarios unidos:

1	2	3	4
0.9209	0.7969	0.5385	0.6782

Como puede verse, a medida que se introducen usuarios, la capacidad de predicción de las emociones diferentes al relax (alegría) se van haciendo más y más difíciles de predecir correctamente. Esto es debido a las diferentes reacciones que experimentan los usuarios ante las pruebas.

Nuestra hipótesis es que, al recoger variables demográficas como la edad, sexo, peso, estado social, altura e introducirlas en el entrenamiento, y con una muestra mucho más grande podría llegar a crearse un modelo genérico bastante eficiente. Pero esta tarea se deja para posteriores investigaciones.

Al realizar este mismo estudio utilizando la red neuronal, se obtuvo el siguiente resultado: con los datos de 5 usuarios unidos, se ha obtenido una tasa de acierto del 98%, para Alegría, 92% para Miedo, 83% para Ira y 87% para

Tristeza, superando con creces los resultados obtenidos mediante el análisis discriminante. Este grafico representa el reparto de los pesos realizado por la Red SOM

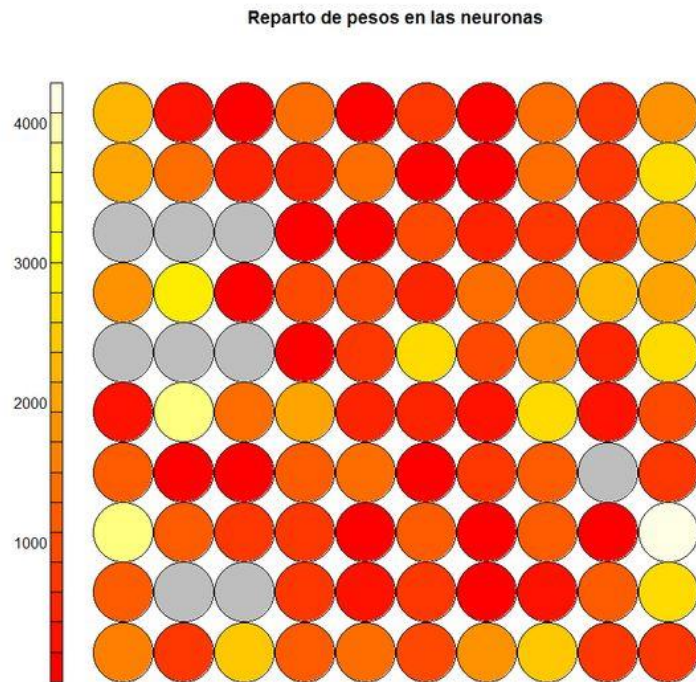


Figura 5.3.1: Distribución de los datos generales con respecto de las neuronas

5. CONCLUSIONES

El objetivo inicial del proyecto era, como se ha detallado a lo largo del documento, construir una herramienta capaz de diagnosticar el estado anímico de un sujeto. Tras nueve meses de proceso de desarrollo, se puede afirmar que se ha desarrollado de forma satisfactoria una aplicación capaz de llevar a cabo dicho objetivo, así como el resto de requisitos (tanto funcionales como no funcionales) mostrados su Sección.

Para llevar a cabo la aplicación, esta se ha dividido en cinco módulos que trabajan de forma independiente: el módulo de recogida y tratamiento de datos, el módulo de la red neuronal, el módulo del servicio web, el módulo Android y el módulo conector. Esta distinción ha dado la posibilidad de estudiar y probar las etapas de forma independiente y así avanzar de forma paralela.

Por un lado, el módulo de recogida y tratamiento de datos ha resultado ser capaz de adquirir los datos de los distintos LOGs y CSVs y transformarlos en datos entendibles para la red neuronal, también resolvió adecuadamente el envío de datos al módulo web.

El módulo de la red neuronal ha conseguido cumplir las expectativas que había depositadas en él consiguiendo, como se relata en el apartado de pruebas, unos muy buenos resultados en la clasificación de emociones mediante los datos biométricos entregados.

El módulo de servicio web nos ha permitido tratar y analizar los datos para poder comprender mediante las analíticas correctas los resultados, así mismo sirvió como backup automático para tener los datos disponibles en cualquier dispositivo en el que se instalase la aplicación.

El módulo Android nos ha permitido portar la lógica a un dispositivo móvil, permitiendo de esta manera que en el futuro si se consigue hacer que toda la información de las dispositivos externos (Shimmer y BodyMedia) se envíe

directamente al móvil con formatos estándar, desvincular la funcionalidad de los ordenadores de sobremesa. También nos aporta una interfaz gráfica que simplifica el uso y ejecución del proyecto.

Por último el módulo conector permite de forma fácil y modular conectar los distintos módulos de la aplicación entre si. De este modo los cambios que hemos sufrido a lo largo del proyecto y los futuros cambios que se realicen en él, se harán de manera rápida e intuitiva.

Con todo esto, la aplicación satisface los objetivos impuestos para el proyecto. Demostrando de esta manera que las emociones de las personas se manifiestan en forma de señales biométricas y estas además siguen patrones definidos para cada una de las persona.

Al realizar el análisis sobre los datos obtenidos hemos comprobado como el cuerpo reacciona de manera muy diferente con cada una de las emociones, estableciéndose así una relación evidente entre las reacciones bioquímicas del cuerpo humano y la emoción que esta sintiendo, siendo así sencillo suponer, con solo unas pocas variables biométricas, el estado emocional del sujeto.

Hay que tener en consideración que nuestro proyecto se ha basado en el estudio de 4 emociones, claramente diferenciadas y de relativa importancia dentro de un contexto biológico. Suponer que al añadir emociones más sutiles, se necesitará una mejora en los algoritmos de detección para no tener una pérdida alta de efectividad.

5.1. Usos de la aplicación

Esta aplicación ha sido desarrollada teniendo en cuenta el campo de la investigación para usos médicos y psicológicos. El hecho de saber si se podría o no detectar el estado anímico de una persona de manera no supervisada es de por

sí solo un gran descubrimiento, pero aparte de poder usarlo para esto el proyecto tiene más campos.

Con pequeños cambios podríamos llegar a ser capaces de detectar patologías médicas y de esta manera contribuir con el campo de la medicina.

La investigación puede ser utilizada, si se focaliza aún más en ciertas emociones, para la detección de la ira, siendo así una ayuda a los casos de control de la ira, agresividad, etc.

También se puede utilizar en el marco del marketing, averiguando las reacciones emocionales de los clientes a un producto de la compañía.

5.2. Futuras ampliaciones

Una vez desarrollada la aplicación, y siendo está completamente funcional, cabe plantearse las posibles ampliaciones que se podrán realizar sobre el sistema para incrementar su funcionalidad y usos. Muchas de ellas parten de la idea inicial del proyecto, con la intención de proporcionar robustez al sistema o mejorar su funcionalidad. Estas ampliaciones son:

- **Ampliar el abanico de emociones detectadas:** Actualmente se han elegido para el proyecto únicamente cuatro emociones que hemos considerado lo suficientemente opuestas para poder llegar a ser clusterizadas. Una vez comprobado que el resultado ha sido el deseado, animamos a que se amplíe la aplicación añadiendo más emociones en ella. Esto debería resultar sencillo de implementar, teniendo en cuenta que la red neuronal sólo tendría que producir un estado más.

- **Ampliar la cantidad de señales biométricas tratadas:** Para el desarrollo del proyecto se han elegido un cierto número de señales biométricas dejando otras fuera de tratamiento ya porque se viera que estaban correladas entre ellas o por la

imposibilidad de acceder a ellas. Por lo tanto una buena y necesaria rama de desarrollo, si se tiene pensado ampliar la cantidad de emociones detectadas, sería ampliar del mismo modo el número de señales biométricas tratadas.

- **Portar el 100% de la aplicación a móvil y servidor:** En estos momentos, la lógica del programa, el almacenaje y el análisis de datos se encuentran alojados en un dispositivo Android y en un servidor, pero la recogida de datos actualmente se ha de hacer mediante ordenador. Esto es debido a que el brazalete no posee cualidades la capacidad de transmitir libremente los datos recogidos de forma inalámbrica y se ha de coordinar la aplicación de recogimiento de datos del shimmer con nuestro proyecto. Debido a esto recomendamos que se investigue el modo de recoger estas señales de forma coordinada con la aplicación.

- **Cambiar emociones por enfermedades:** Aunque nuestro trabajo ha sido para la detección de las distintas emociones, de forma sencilla se podría modificar para la detección de distintas patologías médicas y/o psiquiátricas. Para ello antes habría que hacer un estudio preliminar para comprobar si las patologías que se quieren detectar tienen un impacto biométrico singular o no.

- **Ampliar la funcionalidad** del servidor incorporando una herramienta de visualización B.I. (Pentaho, Jaspersoft, SpagoBI) que permita visualizar cómodamente los datos, realizar informes y dashboards e incorpore accesos de seguridad a los datos.

- **Localizar patrones comunes de bioseñales** en los diferentes usuarios utilizando para ello datos demográficos y personales, para reducir así los tiempos de entrenamiento y estar en disposición de preparar una aplicación que detecte las emociones a múltiples usuarios sin necesidad de realizar un intenso entrenamiento.

5.3. Seguimiento del proyecto

El código fuente de la aplicación móvil y del servidor se puede conseguir para uso exclusivo de investigación mandando un correo a:

jayala@fdi.ucm.es

Para contactar con el grupo de proyecto se puede hacer a través de la siguiente dirección:

emotion-control-proyect-si@googlegroups.com

6. GUIA DE USO

6.1. Introducción.

A continuación mostraremos el manual completo para la instalación y configuración de todas las herramientas utilizadas durante la investigación. De tal forma que el proyecto pueda ser reutilizado y ampliado en posteriores estudios. La guía consistirá en varias partes, separadas por las herramientas necesarias. En primer lugar, describiremos los requisitos necesarios para la instalación de todo el hardware y software necesario. Después, procederemos a explicar la instalación y uso de la aplicación móvil. Por último, explicaremos la instalación y configuración de un servidor externo donde poder almacenar con seguridad los datos recogidos así como los parámetros necesarios para la replicación de las redes neuronales de cada usuario. Además, el servidor incorporará una aplicación para el análisis y estudio de esos datos. Que ayudará a la obtención de conclusiones empíricas de los resultados.

6.2. Requisitos

Para la instalación y configuración del servidor es necesario:

- Un ordenador conectado permanentemente a internet para hacer las tareas de servidor, durante la guía se mostrarán diferentes opciones para obtener esto.
- Un conocimiento nivel usuario de Linux para la ejecución correcta de los comandos necesarios
- Un conocimiento nivel usuario de Apache Tomcat para la configuración personalizada del servidor.
- Un conocimiento nivel medio de la herramienta R y su lenguaje de programación para la realización de análisis.

- Un conocimiento de nivel medio sobre servicios REST y su utilización.

Para la instalación de la aplicación móvil es necesario tener un Sistema Operativo Android con 4.2 óptima o 2.3 mínima.

Para descargar los datos del brazalete se necesitará un script en Python creado por *Centi Benzo* de dominio público. Para ejecutar dicho script, se requiere de un entorno python con diversas librerías de cálculo matricial, por ello en nuestro caso usamos una máquina virtual de linux con la distribución Debian.

6.3. Instalación de la APK

Una vez descargado el APK del programa debemos asegurarnos que podemos instalarlo. Para ello nos dirigimos a "Ajustes" del móvil y en apartado "Seguridad" vamos a "Orígenes desconocidos" y hacemos *click* en el cuadro indicando de esta manera que permitimos la instalación de aplicaciones que no sean descargadas por los métodos estándares.

Una vez que sabemos que tenemos los permisos convenientes, mediante un explorador de archivos nos dirigimos hacia la carpeta que contiene nuestra APK y hacemos *click* sobre ella. A continuación nos preguntará si queremos instalarla y le damos a "Aceptar". Terminado este paso tenemos la aplicación instalada en el móvil.

6.4. Uso de la aplicación móvil

Primero y antes que nada antes de iniciar la aplicación se deben encontrar situados los datos recogidos después del entrenamiento o el diagnóstico en la carpeta correspondiente como se explica más atrás en: CAPTURA DE VARIABLES BIOMÉTRICAS.

Una vez instalada la aplicación móvil nos dirigimos al menú y la ejecutamos. En este momento deberíamos ver la una interfaz como la que se muestra a continuación:



The image shows a screenshot of the JStress mobile application. At the top, there is a black header with the Android logo and the text 'JStress'. Below the header, the text 'Emotion Control' is centered. There are four buttons stacked vertically: 'Guardar Datos', 'Cargar Datos', 'Entrenamionton Inicial', and 'Diagnostico emocional'. Below the buttons, there are five input fields labeled 'Nombre', 'Altura(cm)', 'Peso(Kg)', 'Edad', and 'DNI'. The 'Edad' field has a vertical cursor and a blue underline.

Como podemos ver en la imagen tenemos la opción de guardar los datos (Para el caso en que hayamos entrenado la red neuronal) o cargar unos anteriores, de esta manera no es necesario entrenar la aplicación cada vez que la iniciemos.

Antes de iniciar un entrenamiento o un diagnóstico se deben rellenar los campos especificados siendo únicamente obligatorio el campo del DNI.

La primera vez que iniciemos deberemos entrar en “Entrenamiento Inicial” e ir dando los tiempos en los que empezó y terminó el entrenamiento correspondiente.



Una vez terminado el entrenamiento de la red neuronal volvemos al menú principal y guardamos los datos. La siguiente vez que entremos en la aplicación nos dirigiremos a "Diagnóstico Emocional" tras poner nuestro DNI y tras decir el tiempo en que hemos estado recogiendo datos ir hacia "Visualizar Diagnóstico" donde veremos el resultado correspondiente en un gráfico.



6.5. Instalación y uso del script para descargar datos

Una vez iniciado el entorno Debian para ejecutar la secuencia de comandos y para ejecutarla por primera vez, debimos modificarla, puesto que las últimas distribuciones de linux cargan automáticamente los componentes hardware recién conectados. Por ello y debido a que el script realizaba la misma acción, hubo que retirar dicha carga.

Tras ello, ha de ejecutarse, con el dispositivo de *BodyMedia* conectado, el script con los siguientes argumentos:

Ruta del dispositivo, la ruta donde se montó la unidad.

Fuente de datos, indicando si los datos se cargan desde el dispositivo o desde otra fuente.

Destino de datos, indicando el destino y formato de los datos.

Esta ejecución, nos genera un archivo CSV (Valores Separados por Coma), donde se encuentran los datos recogidos. Por último y para evitar que se acumulen los datos de distintas personas, hay que realizar un borrado. Esto se indica con el parámetro “*clear*”, que borrara la información contenida en el dispositivo.

6.6. Instalación y montaje de los servidores

Para la instalación del servidor caben dos opciones posibles, o bien, la configuración de un servidor privado con las herramientas que a continuación se describirán. O bien mediante la utilización de los servidores en la nube de Amazon AWS. Esta última opción ha sido la utilizada por nosotros y la guía de uso para la instalación de los servicios partirá de ahí. No obstante, si se deseara utilizar un servidor privado, siempre y cuando este esté sobre un entorno Linux, bastaría la configuración de las herramientas así como la instalación del servidor Tomcat para el arranque del servicio.

1. Configuración Inicial del servidor mediante Amazon AWS.

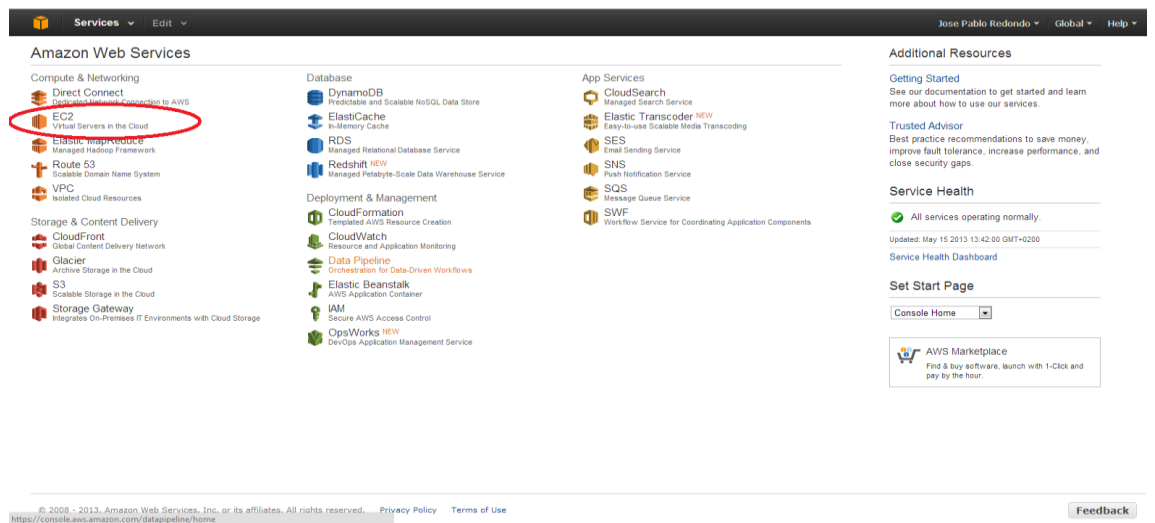
- Amazon, la página web para la compra de libros, posee además un servicio para la creación de servidores en la nube. Este servicio resulta gratuito (con ciertas limitaciones) durante el primer año de uso. Para utilizarlo nos dirigiremos a la página web principal de Amazon AWS⁸.

- El siguiente paso consiste en la creación de una cuenta de Amazon (si aún no se poseyese una). Durante la creación de la cuenta se nos solicitará los datos de una tarjeta bancaria. No se nos cobrará cargo alguno salvo que excedamos las

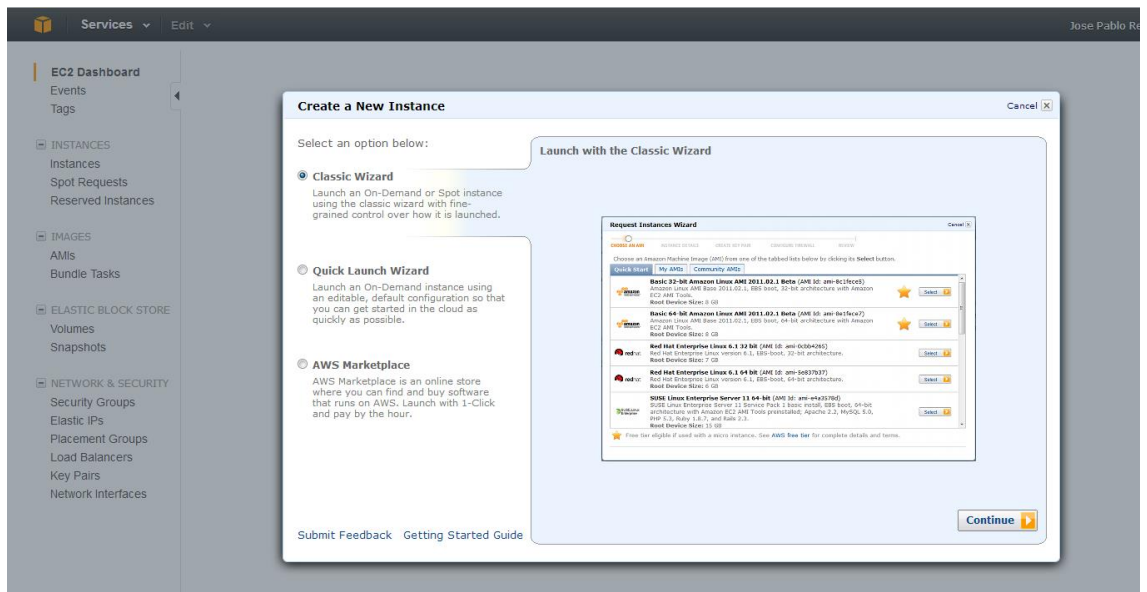
⁸ <http://aws.amazon.com/es/>

limitaciones de la capa de uso gratuito, no obstante podremos configurar alarmas para que se nos avise si en algún momento se nos cobra algo.

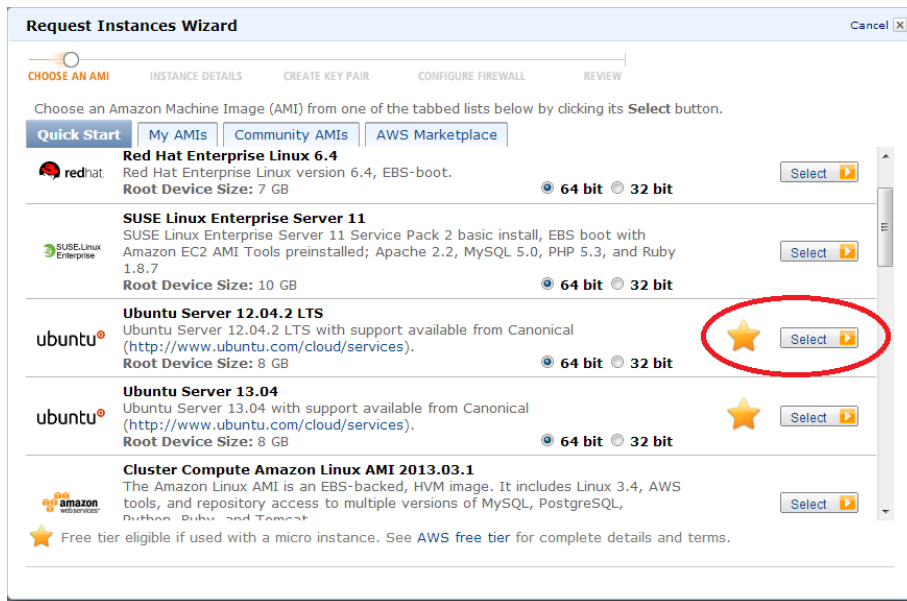
- Una vez la cuenta ha sido creada, nos dirigiremos al AWS Management Console, desde donde podremos configurar todo aquello que necesitemos. En particular, nos dirigiremos a la opción EC2 Dashboard, que es el servicio que controla la creación y gestión de servidores en la nube.



- En la sección Create Instance, seleccionamos Launch Instance y se nos mostrará un Asistente para la creación de servidores. Podemos seleccionar cualquiera de las dos vías, la Express y la clásica, ambas nos permitirán generar el servidor requerido. En particular, nosotros utilizaremos la Opción Clásica.



- Entre las distintas opciones de Sistemas operativos ofrecidos por la nube de amazon seleccionaremos Ubuntu Server 12.04 LTS x64. En teoría es admisible cualquier sistema operativo basado en Debian con apt-get como utilidad de instalación de software. En caso de preferencia, las herramientas pueden también ser instalados en sistemas CentOS y RedHat usando yum, pero esta guía no cubrirá estas instrucciones. Baste decir que con la instalación básica de las herramientas es necesario para que el servicio se active y funcione.



- El resto de las configuraciones del servidor las escogeremos de aquellas que los conceda la capa de uso gratuito de Amazon, CPU: T1 Micro 613 Mb de RAM, creamos una llave de acceso y la almacenamos en aquellos ordenadores que usaremos para acceder a la consola. En cuanto al disco duro, la capa de uso gratuito nos permite tener hasta 30GB de almacenamiento online, por lo que incrementaremos el tamaño del disco duro hasta esa cantidad.

Request Instances Wizard Cancel

CHOOSE AN AMI | **INSTANCE DETAILS** | CREATE KEY PAIR | CONFIGURE FIREWALL | REVIEW

Provide the details for your instance(s). You may also decide whether you want to launch your instances as "on-demand" or "spot" instances.

Number of Instances: 1 **Instance Type:** T1 Micro (t1.micro, 613 MiB)

Launch as an EBS-Optimized instance (additional charges apply): Not supported for this instance type

Launch Instances

EC2 Instances let you pay for compute capacity by the hour with no long term commitments. This transforms what are commonly large fixed costs into much smaller variable costs.

Launch into: EC2-Classical EC2-VPC

Availability Zone: No Preference

Request Spot Instances

[Back](#) [Continue](#)

- Por último, crearemos un grupo de seguridad que incluya los puertos 22, 80 y 8080 abiertos.

Services | Edit | Jose Pablo Redondo | Ireland

Create Security Group **Delete**

Viewing: EC2 Security Groups | Search

Name	VPC ID	Description
emotionSecurity		security group
default		default group

1 Security Group selected

Security Group: emotionSecurity

Details **Inbound**

Create a new rule: Custom TCP rule

Port range: (e.g., 80 or 49152-65535)

Source: (e.g., 192.168.2.0/24, sg-47ad482e, or 1234567890/default)

TCP Port (Service)	Source	Action
22 (SSH)	0.0.0.0/0	Delete
80 (HTTP)	0.0.0.0/0	Delete
8080 (HTTP*)	0.0.0.0/0	Delete

- Una vez confirmado, el servidor se creará y activará automáticamente. La capa de uso gratuito nos da horas suficientes para tener un servidor activado 24/7 así que no será necesario preocuparse de desconectarlo.

2. Instalación de Oracle Java:

- El servicio REST utilizado en el servidor, así como el servidor Tomcat mismo, están basados en Java, por lo que será necesaria la instalación de la versión 7 de Oracle. Las instrucciones para su instalación pueden encontrarse en esta página⁹ indicada.

- Una vez terminada la instalación será necesaria la creación de una variable del sistema para que el servidor de Apache Tomcat sea capaz de localizar las librerías necesarias para su activación.

- Para ello nos dirigimos a la carpeta /etc/profile.d donde crearemos con el editor de texto que queramos (vi y vim vienen preinstalados) un archivo con un nombre cualquiera y la extensión .sh y escribimos lo siguiente:

```
export JAVA_HOME=/usr/lib/jvm/java-7-oracle (o la carpeta donde este
instalado java)
```

```
export CATALINA_HOME=/opt/tomcat (la carpeta donde instalaremos el
tomcat).
```

```
export JAVA_HOME=/usr/lib/jvm/java-7-oracle
export CATALINA_HOME=/opt/tomcat
```

- Guardamos y reiniciamos sesión. Para comprobar que se ha hecho correctamente, podemos escribir **echo \$JAVA_HOME** y el sistema debería devolvernos la carpeta de java.

```
ubuntu@ip-10-48-127-13:~$ echo $JAVA_HOME
/usr/lib/jvm/java-7-oracle
ubuntu@ip-10-48-127-13:~$
```

- Si este método no funcionase, también se pueden escribir estas líneas en el archivo .bashrc localizado en ~/.bashrc

3. Instalación del servidor de Base de Datos:

⁹ <http://www.ubuntu-guia.com/2012/04/instalar-oracle-java-7-en-ubuntu-1204.html>

- Para la base de datos utilizaremos un conector con MongoDB (en posteriores secciones se explicará y comentará el motivo de esta elección). Su instalación es sencilla, y es suficiente con la configuración básica. Se puede encontrar en su página oficial¹⁰:

- Una vez instalado, no sería necesario realizar ninguna otra acción pues el servidor se queda activado una vez se ha realizado la instalación correctamente.

4. Instalación del servidor Apache Tomcat:

- Para la instalación del servidor de Apache podemos utilizar el archivo descargable de su página oficial¹¹. Una vez descargado el archivo lo descomprimos en la carpeta /opt/ y renombraremos la carpeta (que contendrá tomcat más la versión) a simplemente tomcat/ para que coincida con la variable de entorno creada previamente.

- Una vez tomcat este comprimido, movemos el archivo receptor.war, que contiene toda la información del servicio REST así como el conector a MongoDB, a la carpeta /opt/tomcat/webapps/

- Por último accedemos a la carpeta /opt/tomcat/conf y editamos el archivo tomcat-users.xml añadiendo un usuario y una contraseña con el rol: manager-gui. (es necesario crear el rol previamente, usando el mismo esquema que se especifica en el mismo archivo.

¹⁰ <http://docs.mongodb.org/manual/tutorial/install-mongodb-on-ubuntu/>

¹¹ <http://tomcat.apache.org/>

```
ubuntu@ip-10-48-127-13: /opt/tomcat/conf
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<tomcat-users>
<!--
NOTE: By default, no user is included in the "manager-gui" role required
to operate the "/manager/html" web application. If you wish to use this app,
you must define such a user - the username and password are arbitrary.
-->
<!--
NOTE: The sample user and role entries below are wrapped in a comment
and thus are ignored when reading this file. Do not forget to remove
<!-- ... --> that surrounds them.
-->

<role rolename="tomcat"/>
<role rolename="role1"/>
<role rolename="manager-gui"/>
<user username="jpadmin" password="XXXXXXXXXX" roles="manager-gui"/>

/tomcat-users>
```

5. Instalación de la herramienta de análisis R:

- Para la instalación de la herramienta R podemos seguir fácilmente los pasos especificados en la página indicada¹².

- Para la ejecución de algunas de las opciones del servidor (aún por implementar) para que trabaje directamente sobre R mediante órdenes REST, será necesario la instalación del paquete RServe. Para ello, entramos en el prompt de R con el comando: R y escribimos `install.packages("RServe")`. Una vez terminado, bastará con escribir `RServe()` en el prompt de R para arrancar el servidor remoto.

6. Arranque del sistema:

- Una vez todas las herramientas instaladas, se puede proceder al arranque de todo el sistema dirigiéndonos a la carpeta de Tomcat: `"/opt/tomcat/bin/"` y arrancando el servidor con el comando: `"/startup.sh"`.

7. Posibles problemas:

- Podría darse la situación de que el servidor no se desplegara correctamente.

¹² http://craig-russell.co.uk/2012/05/08/install-r-on-ubuntu.html#_UY5f6LXKGZc

- En ese caso, deberemos dirigirnos a la página web de nuestro servidor (que por defecto corresponderá a la dirección IP más el puerto con el formato xxx.xxx.xxx.xxx.:8080/ y acceder al manager-gui con la contraseña y usuario que creamos anteriormente. Desde ahí puede lanzarse manualmente el servicio.
- Si este siguiese sin arrancar, se deben comprobar el log catalina.out, situado en la carpeta “/opt/tomcat/logs/”

6.7. Información sobre las frases, videos, etc, para la inducción de emociones

A la hora de inducir emociones en el usuario poseemos una serie de frases, músicas y videos que nos permitirán que el usuario experimente las emociones que nosotros queremos analizar.

Las frases, los videos y la música se encuentran mencionados en el Apéndice.

Aparte de los videos y frases lo primero que vamos a necesitar es una sala, con un ordenador, en la cual podamos nivelar la cantidad de luz que entra (ya sea mediante una lámpara o una persiana) y que esté relativamente bien aislada de sonidos externos. También es aconsejable que el usuario se sienta en una silla.

Según se expone más adelante el usuario deberá leer una frases en ciertas pruebas, para que el resultado sea el deseado se debe explicar al usuario que debe leer las frases con tranquilidad y pronunciarlas en voz alta a la vez que de fondo tiene la música ambiental adecuada. En el momento de ver el video el ambiente de la sala deberá ser propicio (En el caso del miedo a de dejarse la sala sin luz) y explicarle al usuario que debe estar atento a lo que suceda.

En ningún momento el usuario debe saber de antemano que frases o videos se le van a enseñar a continuación.

7. TECNOLOGÍAS USADAS

Oracle Java SE 7.0¹³.



Todas las herramientas utilizadas en el proyecto han sido desarrolladas utilizando la última versión de Java Oracle.

La razón de esta decisión ha sido debida a la propia naturaleza del proyecto, que implicaba la realización de una aplicación Android, para cuyo desarrollo Java es la tecnología recomendada. Una vez tomada esa decisión, el resto de herramientas se desarrollaron también en Java por razones de consistencia. Además, Java ha permitido el uso de nuevas tecnologías en forma de Frameworks que han ayudado a poder desarrollar rápidamente cualquier necesidad que el proyecto pudiera requerir en un momento determinado.

Apache Tomcat 7¹⁴:



¹³ <http://www.oracle.com/technetwork/es/java/javase/overview/index.html>

¹⁴ <http://tomcat.apache.org/>

El servidor web que ofrece el servicio de almacenamiento está levantado sobre un servidor Tomcat.

Tomcat es una solución Open Source desarrollada por la fundación Apache para la implantación de servicios Web mediante Java.

Sus características Open Source (en contraste con alternativas como JBoss) así como su fácil funcionamiento utilizando Java y documentos xml para su configuración han sido los factores decisivos para su elección.

MongoDB¹⁵:



MongoDB es un sistema de bases de datos NoSQL basado en el almacenamiento y extracción de documentos. Es un sistema Ágil y rápido, así como escalable, lo que lo hace ideal para bases de datos con posibilidades de crecer exponencialmente. Algunas de sus principales características relatadas en su página oficial que se tuvieron en consideración para su elección, son las siguientes:

- Es un sistema orientado al almacenamiento de documentos. Lo que permite el parseo directo y su almacenamiento de archivos JSON.

¹⁵ <http://www.mongodb.org/>

- Capacidad de Indexación Total: MongoDB permite crear índices sobre cada uno de atributos que uno desee.
- Alta Disponibilidad y Replicado: MongoDB posee un sistema de replicado de información a través de LAN y WAN, permitiendo un acceso rápido y garantizado.
- Auto-Sharding: MongoDB escala horizontalmente, dividiendo su base de datos en diferentes particiones, y obteniendo la información sin pérdida de velocidad a un menor coste.
- Open Source: Es un sistema completamente gratuito, abierto y con una fuerte comunidad detrás.
- Querying: Mongo permite la ejecución de queries sencillas directamente sobre los documentos.

Todas estas características fueron clave para la decisión de su utilización en lugar de otras opciones más comúnmente conocidas como MySQL. Nuestro sistema funciona mediante el envío de archivos JSON al servidor para su almacenamiento. Para evitar un parseo innecesario, se usa MongoDB ya que posee la capacidad de almacenarlos directamente. Además, dada la naturaleza de la información recolectada (señales de sensores) es de prever que la base de datos crezca de manera que sea necesaria la utilización de sistemas mejor preparados para trabajar con una mayor cantidad de información. Por último, al ser este un proyecto de investigación, es razonable que las características de los datos cambien, se añadan o se retiren señales o información dependiendo de los resultados que se vayan obteniendo. Un sistema basado en documentos permite una mayor libertad al poder añadir más documentos sobre una colección concreta aunque estos posean más o menos características que los que hubiera previamente.

Además, aunque está basado y creado en C++, posee un conector que permite su utilización directamente desde Java.

The R project¹⁶:



R es un paquete de herramientas estadísticas y gráficas desarrollado para que sea completamente Open Source. Una de sus principales características es la enorme comunidad que posee detrás, generando librerías constantemente, hasta el punto de que R posee actualmente cientos y cientos de librerías que permiten realizar cualquier tarea de Machine Learning, estadística, análisis matemático y gráfico que se pueda necesitar.

Todo ello con un código completamente abierto y sencillo. Teniendo su propio lenguaje. Es el sustituto perfecto para Matlab, cuya utilización no es en absoluto gratuita. Además, entre muchas de sus librerías, R posee conectores a Java, para permitir la ejecución, tanto de scripts, como de código directamente desde aplicaciones externas, otra de sus librerías utilizada es Rserve, que permite una conexión remota desde un ordenador exterior para la ejecución, mediante las GUIs disponibles (RCommander¹⁷ o RStudio¹⁸, de cualquier análisis necesario.

RESEasy¹⁹:



Según la propia definición de la Wikipedia, La **Transferencia de Estado Representacional** (Representational State Transfer) o **REST** es una

¹⁶ <http://www.r-project.org/>

¹⁷ <http://www.rcommander.com/>

¹⁸ <http://www.rstudio.com/>

¹⁹ <http://www.jboss.org/reteasy>

técnica de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web. En la actualidad se usa en el sentido más amplio para describir cualquier interfaz web simple que utiliza XML y HTTP, sin las abstracciones adicionales de los protocolos basados en patrones de intercambio de mensajes como el protocolo de servicios web SOAP.

Dentro de las diferentes implementaciones que se han diseñado para los Servicios REST, se ha escogido RESTEasy debido a que se encontró mejor documentación sobre cómo utilizarlo y programarlo.

Apache Maven²⁰:



Maven es una herramienta Open Source para la gestión de librerías y construcción de proyectos Java, mediante la utilización de documentos XML. Lo que lo hace bastante sencillo de manejar.

Maven se encarga de mantener actualizadas las dependencias del proyecto mediante sus repositorios. Debido a la variedad de tecnologías utilizadas en la creación del servidor Web, Maven ha sido la solución más sencilla para gestionarlas todas y mantenerlas funcionales de manera sencilla.

Spring:



²⁰ <http://maven.apache.org/>

Spring es el framework más utilizado en estos momentos para el desarrollo en Java, permite un sistema de inyección de dependencias en las clases Java, creando así una aplicación reusable mediante interfaces, fácilmente testeable con JUnit, y totalmente preparado para su ejecución rápida.

RStudio:

RStudio es un IDE libre y de código abierto para R, un programa para estadística computacional y graficas.

GIT:

Git es un software de control de versiones, pensado para la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente.

SDK de Android

El SDK de Android, incluye un conjunto de herramientas de desarrollo. Comprende un depurador de código, biblioteca, un simulador de teléfono basado en QEMU, documentación, ejemplos de código y tutoriales. La plataforma integral de desarrollo (IDE, Integrated Development Environment) soportada oficialmente es Eclipse junto con el complemento ADT (Android Development Tools plugin), aunque también puede utilizarse un editor de texto para escribir ficheros Java y Xml y utilizar comandos en un terminal (se necesitan los paquetes JDK, Java Development Kit y Apache Ant) para crear y depurar aplicaciones. Además, pueden controlarse dispositivos Android que estén conectados.

8. APÉNDICE

A continuación enumeraremos las distintas canciones que se aconsejan para inducir los distintos estados anímicos:

Estado de relajación:

- Beethoven, Sinfonía N° 3 "Eroica"
- Four Seasons ~ Vivaldi

Estado de depresión:

- Beethoven Sonata No. 8 'Pathetique' Mov. 2
- Beethoven - Moonlight Sonata
- Atrium Carceri - End Titles dark ambient

Estado de ira:

- Alkan Concerto for Solo Piano - First movement
- O Fortuna ~ Fortune plango vulnera
- Mozart " Requiem Dies Irae "

En el caso del miedo no habrá ni música ni frases, únicamente habrá un video que ha de ser un montaje con escenas de terror y que contengan sobresaltos.

Por último mostraremos las frases utilizadas para la inducción de los diferentes estados anímicos.

Estado de relajación:

- 1.- Hoy no es ni mejor ni peor que cualquier otro día.
- 2.- No obstante, hoy me siento bastante bien.
- 3.- Siento que me he quitado un enorme peso de encima
- 4.- Este puede haberse convertido en uno de mis mejores días.

5.- Si tu actitud es positiva, entonces las cosas son maravillosas y yo tengo una actitud positiva.

6.- Me siento vivo y animado.

7.- Ciertamente he logrado energía y confianza en mí mismo para compartir.

8.- En general, tengo poca dificultad en pensar de manera clara y sencilla.

9.- Mis amigos y familia están muy orgullosos de mí la mayor parte del tiempo.

10.- Estoy en buena posición para hacer que las cosas salgan adelante.

11.- Para lo que queda de día, me apuesto lo que sea que las cosas me van a ir de perlas.

12.- Estoy agradecido de que la mayoría de las personas sean amigables conmigo.

13.- Mis juicios sobre la mayoría de las cosas son sonoros.

14.- Cuánto más me meto en algo, más fácil me resulta.

15.- Estoy lleno de energía y ambición - Siento que puedo ir sin dormir durante un buen rato.

16.- Este es uno de esos días cuando puedo conseguir las cosas sin prácticamente esfuerzo.

18. Cuando quiero, puedo hacer amigos fácilmente.

19. Si me concentro, puedo hacer las cosas bien.

20. Ahora me siento entusiasmado y seguro.

21. Deben venir buenos tiempos.

22. Pasan por mi cabeza mis canciones preferidas

23. Mis amigos están muy animados y optimistas

24. Me siento hablador - Siento como si pudiera hablar con cualquiera.

25. Estoy lleno de energía, estoy empezando a disfrutar las cosas que hago.

26. Siento como si fuera a explotar en carcajadas - Me gustaría que alguien contase un chiste y me diera una excusa para reír

27. Siento una animación estimulante en todo lo que hago

29. Soy capaz de hacer las cosas con precisión y eficiencia

30. Se muy bien que puedo alcanzar las metas que me he puesto

31. Ahora que me doy cuenta, casi todas las cosas que me han deprimido no lo hubieran hecho si hubiera tenido la actitud correcta
33. Me siento tan vivo como si estuviera en la cima del mundo
34. Sería muy difícil detenerme ahora
35. A la larga, es obvio que las cosas han ido mejorando a lo largo de mi vida
36. Se que en el futuro no voy a sobrevalorar los llamados problemas
37. Soy optimista porque me puedo llevar bien con casi todos los que conozco
38. Estoy demasiado ocupado para tener tiempo para preocuparme
39. Me siento increíblemente bien hoy
40. Estoy particularmente inventivo y lleno de recursos de este humor
41. ¡Me siento genial! Creo que puedo trabajar dando lo mejor de mi
42. Las cosas pintan bien, ¡Las cosas pintan genial!
43. Muchas de mis amistades seguirán conmigo en el futuro
44. Me siento muy perceptivo y fresco
45. Puedo encontrar lo bueno en casi todo
46. Estoy super animado, puedo hacerlo todo rápido y a la primera
47. Me puedo concentrar completamente en todo lo que hago
48. Mi mente esta clara y rápida
49. La vida es muy divertida, ofrece muchas cosas para sentirse completo
50. Las cosas irán a mejor y mejor hoy
51. Puedo tomar decisiones rápido y correctamente y puedo defenderlas de críticas fácilmente
52. Me siento endemoniadamente productivo ¡Quiero algo que hacer!
53. La vida está firmemente bajo mi control
54. ¡Me gustaria que alguien pusiera buena música a buen volumen!
55. Esto es genial, me siento realmente bien. ¡estoy eufórico con todo!
56. Me siento realmente agudo
57. ¡ Este es uno de esos días en los que estoy listo para todo!
58. ¡Me siento genial!

Estado de depresión:

- 1.- Soy una mierda
- 2.- Nada de lo que he hecho ha servido para nada
- 3.- Me siento como si todo fuera en vano
- 4.- Las responsabilidades que tengo me abruman
- 5.- Nada de lo que pueda hacer va a mejorar mi situación
- 6.- Estoy empezando a pensar que no valgo nada
- 7.- Nadie piensa nunca en mi
- 8.- El pesimismo es bueno, así nunca me sorprenderán las cosas malas
- 9.- Todo puede ir a peor siempre
- 10.- No hay nada por lo que merezca la pena luchar
- 11.- Estoy cansado de todo
- 12.- Desde aquí solo veo nubes negras
- 13.- El futuro es incierto
- 14.- No creo que pueda pasar nada bueno
- 15.- Ojala sucediera algo bueno alguna vez
- 16.- No tengo esperanza en nada
- 17.- No merece la pena
- 18.- ¿Para qué esforzarse?
- 19.- Todo va a dar igual, siempre da igual
- 20.- El mundo va a peor, cada día empeora
- 21.- Hoy es peor que ayer, pero mejor que mañana
- 22.- Me siento tan solo
- 23.- Nadie me comprende
- 24.- Las preocupaciones me abruman
- 25.- Nadie me quiere
- 26.- Realmente estoy solo
- 27.- ¿Porque me sale todo mal?
- 28.- ¿Que estoy haciendo con mi vida?

- 29.- Todo lo que hago es inútil.
- 30.- Estoy cansado de todo esto.
- 31.- Aunque lo intentara se que fallaría
- 32.- Nadie me esta esperando
- 33.- Tengo demasiado trabajo
- 34.- Esta vida es una mierda
- 35.- No valgo para nada
- 36.- Nada me sale bien
- 37.- Si me muero...¿le importara a alguien?
- 38.- Ojala mi vida fuera de otro
- 39.- ¿Para qué esforzarse si no seré capaz de ser feliz?
- 40.- Nadie me entiende, estoy solo en este mundo
- 41.- Todo lo malo me pasa a mí
- 42.- Si crees que estás haciendo una cosa bien, es porque no te estás dando cuenta de que estás equivocado
- 43.- Hubiera sido mejor que no hubiera nacido

Estado de ira:

- 1.- ¿Como han sido capaces de ello?
- 2.- No tienen ni puta idea
- 3.- Son gilipollas
- 4.- Se merecen lo peor
- 5.- No voy a perdonar eso nunca
- 6.- No voy a aguantarlo más
- 7.- Estoy harto
- 8.- ¿Pero que mierdas se han creído?
- 9.- No me pueden tratar así
- 10.- Se van a enterar
- 11.- Esto es una injusticia
- 12.- No me hace ni puta gracia
- 13.- No voy a aguantar ni una más
- 14.- Les voy a partir las piernas
- 15.- Y luego vendrán a quejarse de sus mierdas
- 16.- ¿Por qué me miras mal?
- 17.- ¿Es que crees que soy imbécil?
- 18.- Hijo de puta lo serás tú
- 19.- No soy gilipollas, no me voy a creer esa tontería
- 20.- Vete a la mierda
- 21.- Te lo mereces por putearme
- 22.- No te atrevas a repetirlo
- 23.- ¿Se creen mejores que yo?
- 24.- Deja de molestar
- 25.- Ya estoy cansado de ti y de los tuyos
- 26.- Si vuelves a dirigirme la palabra no respondo a mis acciones
- 27.- YA BASTA
- 28.- YA ES HORA DE CAMBIAR LAS COSAS

- 29.- ESTO NO SEGUIRA ASI POR MUCHO TIEMPO
- 30.- NO QUIERO VOLVER A VERTE
- 31.- SI SUPIERAS LO QUE YO SE NO TE REIRÍAS TANTO
- 32.- SIEMPRE QUEJÁNDOSE
- 33.- Yo aguanto mil veces más sin quejarme
- 34.- Siempre me habéis tenido por tonto, pero nunca más.
- 35.- ¿CREES QUE ESTO SE ACABA AQUÍ?
- 36.- Te odio y siempre te he odiado
- 37.- Ahora a quien toca sufrir es a ti.
- 38.- Hijo de puta
- 39.- Me cago en la puta
- 40.- Ojalá se mueran
- 41.- Me arrepiento de haberles conocido
- 42.- ¡Dejadme en paz!