
UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

MÁSTER UNIVERSITARIO EN LENGUAJES Y SISTEMAS INFORMÁTICOS

TECNOLOGÍAS DEL LENGUAJE EN LA WEB

Asignatura Minería de Datos

Trabajo Fin de Master

# MINERÍA DE DATOS SOBRE ESTRELLAS ENANAS ULTRAFRÍAS

Autor: CÉSAR CARRIÓN ROSILLO

Dirigido por: LUIS MANUEL SARRO BARO

Curso: 2010 – 2011


# MINERÍA DE DATOS SOBRE ESTRELLAS ENANAS ULTRAFRÍAS

Realizado por: CÉSAR CARRIÓN ROSILLO

Dirigido por: LUIS MANUEL SARRO BARO

Tribunal calificador:

Presidente: D./Dª. .....

Secretario: D./Dª. . . . . .

Vocal D./Dª. . . . . .

Fecha de lectura y defensa: . . . . . .

Calificación .... . . . . .

## Título

# MINERÍA DE DATOS PARA ESTRELLAS ENANAS ULTRAFRÍAS

## Resumen

Gaia es una misión de la Agencia Espacial Europea que se propone con el objetivo de realizar el censo más extenso y detallado de estrellas de nuestra Galaxia (la Vía Láctea). Gaia observará alrededor de mil millones de objetos proporcionando para cada uno de ellos, la característica más difícil de medir en astrofísica la distancia. Además, proporcionará los movimientos propios de cada estrella, información sobre su espectro y sobre los parámetros físicos que los caracterizan

El Análisis y Procesamiento de tal cantidad de datos es un reto en sí mismo, que pondrá en juego las más avanzadas tecnologías de todo tipo, incluyendo las del área del aprendizaje automático y la minería de datos

Dicho análisis se llevará a cabo por un consorcio internacional denominado DPAC en el que participa el departamento de Inteligencia Artificial de la UNED

El DPAC se divide en 8 unidades de coordinación. En la octava dedicada a la extracción de parámetros físicos a partir de los espectros es donde se enmarca el presente PFC

En el presente TFM se propone localizar y modelizar el mejor algoritmo de predicción de la temperatura efectiva sobre las estrellas Enanas Ultrafrías, partiendo de un conjunto de datos simulados que deben ser similares a los datos obtenidos por la sonda GAIA.

**Lista de palabras claves**

ESA, Gaia, DPAC, CU, estrellas enanas ultrafrías, wavelets, moving average, Procesos Gausianos, Máquinas de Vectores Soporte, Regresiones de base radial, K-vecinos cercanos, KNN, normalización euclídea, normalización areal, weka

Title

# DATA MINING FOR ULTRA COOL DWARFS


## Abstract

Gaia is a mission of European Space Agency proposed in order to perform more extensive and detailed survey of stars in our Galaxy (the Milky Way). Gaia will observe about one billion objects providing for each of them the property more difficult to measure in astrophysics: the distance. It will provide the proper motions of each star, information on their spectra and physical parameters.

Analysis and Processing of such amount of data is a challenge in itself, which will play the most advanced technologies of all kinds, including the area of machine learning and data mining

This analysis will be conducted by an international consortium called DPAC (the acronym of Data Processing and Analysis Consortium), which involved the UNED Department of Artificial Intelligence. The DPAC is divided into eight units of coordination. It is in CU8, dedicated to the extraction of physical parameters from the spectra, that this TFM is framed.

The algorithms that perform the task of predicting the values of a function from its variables are called regression algorithms. In this TFM we propose to implement and modelize best effective temperature prediction algorithm, using data set simulated as data set obtained by GAIA.



**Key words**

ESA, Gaia, DPAC, CU, ultracold dwarf star, wavelets, moving average, Gaussian Processes, Support Machine Vectors, RBF regresions, KNN, K-Nearest Neirbourgh, euclidean normalization, areal normalization, weka

INDICE

**Siglas, Abreviaturas y Acrónimos**

| | |
|---|---|
| AP | Astrophysical Parameters |
| BP | Blue Photometer |
| CU | Coordination Unit (in DPAC) |
| DPAC | Data Processing and Analysis Consortium |
| ECM | Error Cuadrático Medio |
| ESA | European Space Agency |
| ESP | Extended Stellar Parametrizer |
| GST | Gaia Science Team |
| GP | Gaussian Processes |
| K-NN | K – Nearest Neighbor |
| LTE | Equilibrio Termodinámico Local |
| PCA | Principal Components Analisis |
| PLS | Partial Least Square |
| RBF | Radial Based Factor |
| RP | Red Photometer |
| SMO o SVM | Máquina de Vectores Soporte |
| UCD | Ultra Cool Dwarfs |

## 1. La ESA y la UNED

### 1.1 La sonda Gaia

Gaia es el nombre propuesto por la Agencia Espacial Europea (en adelante ESA) para una sonda espacial, sucesora de la misión Hipparcos, que será lanzada en la segunda mitad del 2011 con fines astrométricos

Esta misión esta incluida dentro del contexto del programa científico de largo plazo de la ESA llamado Horizon 2000

Gaia se situará en una órbita de Lissajous, alrededor del sistema Sol-Tierra, en el punto L2 de Lagrange. Este punto se encuentra en la línea formada por dos masas (Sol-Tierra) y más allá de la menor de ellas

La sonda Gaia obtendrá un catálogo de aproximadamente mil millones de estrellas de hasta magnitud aparente 20

Sus objetivos comprenden: (a) medidas astrométricas (o posicionales), determinando las posiciones, distancias y movimiento propio anual de las estrellas, con una precisión de unos 20 μas (microsegundos de arco) a magnitud 15, y 200 μas a magnitud 20; (b) medidas fotométricas, obteniendo observaciones multicolor y multiépoca de cada objeto detectado, y (c) medidas de velocidad radial

Gaia creará así un mapa tridimensional extremadamente preciso de las estrellas de nuestra galaxia, la Vía Láctea. También hará un mapa de sus movimientos dando pistas sobre el origen y evolución de la Vía Láctea

Las medidas fotométricas realizadas se usarán para obtener las propiedades físicas detalladas de cada estrella observada, caracterizando su luminosidad, temperatura, gravitación, y la composición de elementos químicos.

Este masivo censo estelar proporcionará los datos básicos necesarios para abordar un amplio rango

de problemas relacionados con el origen, estructura y evolución e historia de nuestra Galaxia Simultáneamente, permitirá la medición de un gran número de quasares, galaxias, planetas extra solares y cuerpos del sistema solar

Cada una de las estrellas sera monitorizada alrededor de 70 veces en un periodo de 5 años calculando sus posiciones, distancias, movimientos y cambios en luminosidad. Se espera descubrir cientos de miles de nuevos objetos celestes como planetas extra solares y enanas marrones

Dentro de nuestro Sistema Solar, Gaia identificará, también, millones de asteroides

Además ofrecerá nuevas pruebas sobre la Teoria de Relatividad General de Albert Einstein

### 1.2 El viaje

Gaia orbitará alrededor del Sol a una distancia de 1,5 millones de kilómetros de la Tierra. Esta localización especial conocida como punto de Lagrange L2, seguirá su camino con la órbita de la Tierra alrededor del Sol.

Observará las estrellas desde esta posición, que ofrece un ambiente térmico estable, con una eficiencia de observación altísima y una radiación moderada (ya que el Sol, la Tierra y la Luna estarán detrás de los instrumentos de observación).

Se ha planificado un tiempo de vida operacional de cinco años.

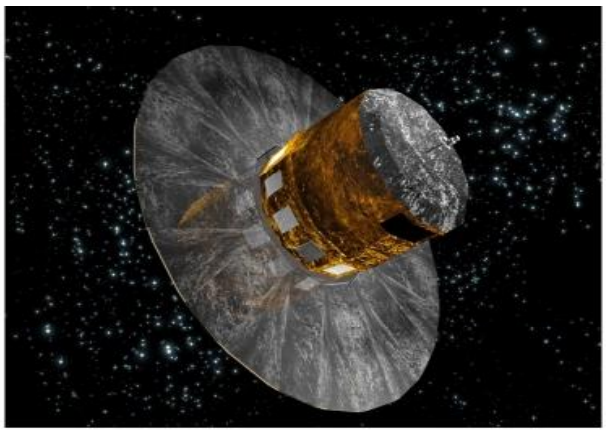

Figura 1: Sonda espacial Gaia

### 1.3 El consorcio de análisis y procesamiento de los datos (DPAC)

La comunidad científica del satélite GAIA está compuesta por un amplio y variado grupo de científicos que han apoyado la misión antes de su selección y que usarán los productos generados por GAIA para sus investigaciones

Un subconjunto de esta gran comunidad está relacionada con el procesamiento de datos, esto es, que toma parte en las fases de desarrollo, que proporcionan tareas de dirección, algoritmos, software o son miembros de cualquiera de los centros de proceso de datos (Data Processing Centres o DPCs).

Este grupo se ha organizado en un consorcio internacional llamado Consorcio de Análisis y Procesamiento de Datos (Data Processing and Analysis Consortium o DPAC).

El DPAC ha establecido una organización interna, una estructura de dirección y garantiza la disponibilidad del hardware necesario para cumplir con las diversas tareas

El DPAC se compone de un conjunto de unidades de coordinación (CU o Coordination Units), cada una de las cuales es responsable de un aspecto clave del procesamiento de los datos

Cada CU, a su vez, puede subdividirse en grupos más pequeños llamados unidades de desarrollo (DU o Development Units) que están a cargo de uno o más paquetes de trabajo

Las CUs están apoyadas por un conjunto de centros de proceso de datos (Data Processing Centres o DPCs) y la coordinación global necesaria es realizada por el consorcio ejecutivo (DPACE)

Las distintas unidades de coordinación tienen como objetivo principales los siguientes:

CU1 Arquitectura del sistema

CU2 Simulaciones de datos

CU3 Procesamiento Core

CU4 Procesamiento de objetos

CU5 Procesamiento fotométrico

CU6 Reducción espectroscópica

CU7 Procesamiento de la variabilidad

CU8 Parámetros astrofísicos

CU9 Acceso al catálogo y exploración científica

En líneas generales el DPAC es responsable de: preparar los algoritmos de análisis de datos para integrar los datos astrométricos, fotométricos y espectroscópicos dentro de un entorno integrado y coherente, incluyendo objetos particularmente característicos como son las estrellas múltiples y planetas menores (minor planets)

- Generar datos simulados para ayudar al diseño, desarrollo y pruebas del sistema global de procesamiento de datos .
- Diseñar, desarrollar, suministrar y operar el entorno hardware y software necesarios para procesar los datos de misión, durante las fases de simulación, operación y producción del catálogo final.
- Diseñar, desarrollar y operar la base de datos final de GAIA, que contendrá los productos, tanto intermedios como finales, que sean de interés para la comunidad científica.

### 1.4 La Unidad de Coordinación 8: Parámetros Astrofísicos

La Unidad de Coordinación de Parámetros Astrofísicos es responsable de las tareas de clasificación Estas tareas usan fotometría, espectrometría y astrometría completamente calibrada para clasificar objetos y estimar sus parámetros astrofísicos.

Los principales objetivos de la clasificación son los siguientes:

1. Clasificación Discreta Determinar cuando un objeto es una estrella, una galaxia, un quasar o un asteroide etc.

2 Estimación de Parámetros Astrofísicos Para los objetos identificados como estrellas, hay que determinar sus propiedades físicas intrínsecas.

Unas de las propiedades más relevantes y obtenibles son la temperatura efectiva (teff), superficie gravitatoria, metalicidad [Fe/H], y la extinción interestelar a lo largo de la línea

de visión (Av), aunque esta última no es, por supuesto, intrínseca a la estrella, lo ideal sería determinar estrella a estrella, aunque se puede considerar como tal

3 Identificación de binarias no resueltas. La mayoría de las estrellas se encuentran en sistemas múltiples

Algunas de estas pueden ser reconocidas por la astrometria, serán las binarias visuales, pero la mayoria no se detectan de esta manera, pero, con relaciones de brillo favorable, puede ser detectada una binaria.

4. Identificación de nuevos tipos de objetos Gaia debe estar abierto a la posibilidad de detectar nuevos tipos de objetos estrellas variables, estrellas poco comunes (por ejemplo, breves fases de la evolución estelar), patrones anormales de la abundancia o sistemas múltiples

Los métodos de clasificación supervisada que se utilizan comúnmente para la determinación de parámetros estelares de espectros generalmente obligan a clasificar a los nuevos tipos de objetos en clases de pre-existentes. Los nuevos objetos, por tanto, no se detectan (y las muestras de los tipos conocidos de los objetos se contaminan)

Para más detalles sobre los objetivos, ver la contribución a la conferencia de Paris 2004 sobre el procedimiento de clasificación y la determinación de parámetros estelares

La estructura de las tareas de clasificación se resume en el ICAP-CBJ-019 (disponible en Livelink o el sitio web del ICAP).

### 1.4.1 Estructura de la CU8

La CU6 proporciona los datos de GAIA preparados a la CU8, y ésta le devuelve las características astrofísicas de las estrellas, una información que sirve al DPAC para clasificar las mismas

El objetivo del CU8 es clasificar y determinar los parámetros astrofísicos de todas los astros que observa GAIA. El presente trabajo, se encarga de un grupo de estos astros: las enanas ultrafrías. Para ello se apoya en métodos clásicos de regresión

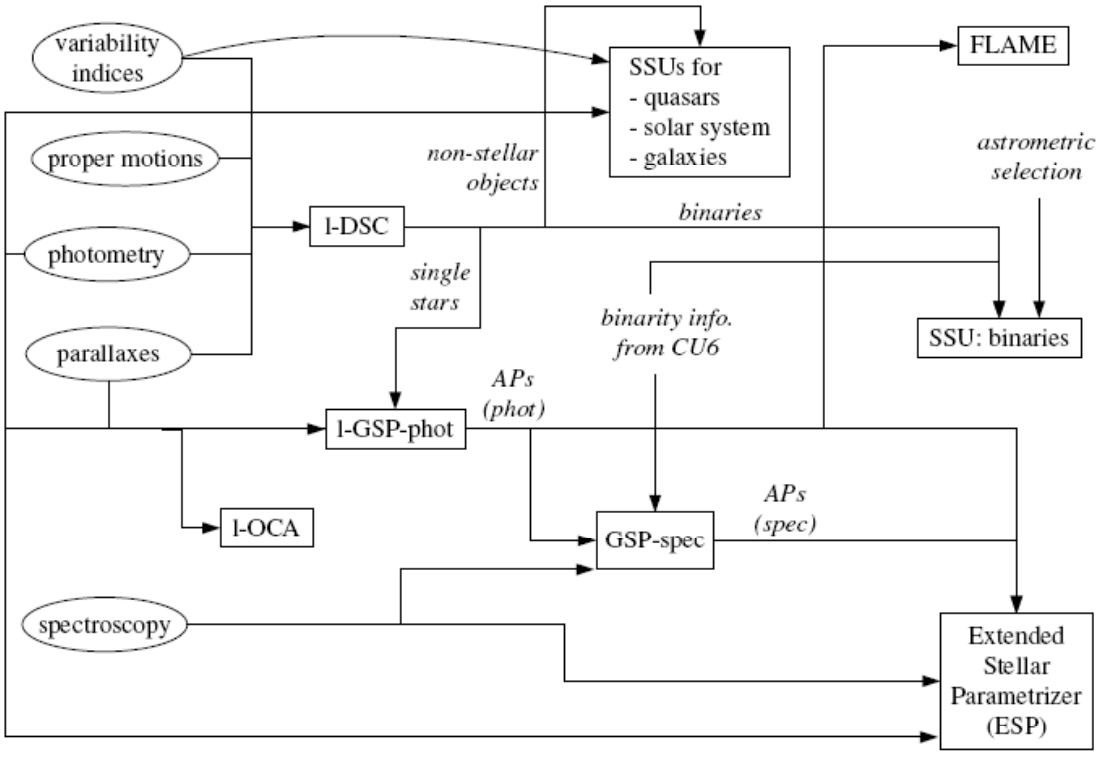


Figura 3: Ubicación del Trabajo dentro de la CU8

El presente estudio en enanas ultrafrías (Ultra Cool Dwarfs) se engloba dentro del área "Extended Stellar Parametrizer (ESP)" del CU8, denominándose "ESP_UCD".

### 1.4.2 Interfaces con otras CU,s

Es muy importante establecer y delimitar las relaciones entre las distintas CU's

Las conexiones con CU7 son muy importantes ya que CU8 proporciona los parámetros estadísticos de las estrellas a CU7, necesario para la clasificación de variabilidad

Según un acuerdo alcanzado con el Gaia Science Team (GST), se publicarán las novedades encontradas que puedan ser de interés general para el resto de la comunidad científica

En el siguiente gráfico se muestra la relación entre las distintas unidades de coordinación, el flujo de datos y los centros de proceso.

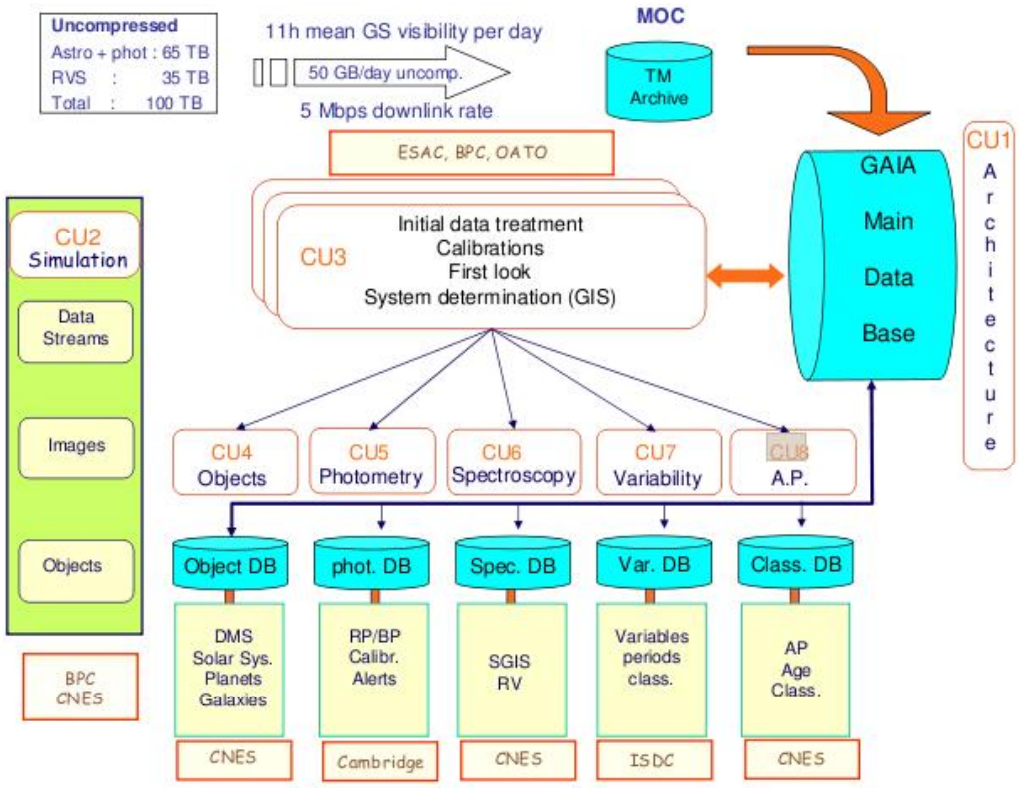


Figura 3 Relación entre las CUs

## 2. El Proyecto: Introducción

### 2.1 Introducción

Gaia es una misión de la Agencia Espacial Europea que se propone con el objetivo de realizar el censo más extenso y detallado de estrellas de nuestra Galaxia (la Vía Láctea)

Gaia observará alrededor de mil millones de objetos proporcionando para cada uno de ellos, la característica más difícil de medir en astrofísica: la distancia

Además, proporcionará los movimientos propios de cada estrella, información sobre su espectro y sobre los parámetros físicos de sus superficies (temperatura, metalicidad, gravedad...)

El Análisis y Procesamiento de tal cantidad de datos es un reto en sí mismo, que pondrá en juego las más avanzadas tecnologías de todo tipo, incluyendo las del área del aprendizaje automático y la minería de datos

Dicho análisis se llevará a cabo por un consorcio internacional denominado DPAC (las siglas en inglés del Consorcio de Análisis y Procesamiento de Datos de Gaia) en el que participa el departamento de Inteligencia Artificial de la UNED.

El DPAC se divide en 8 unidades de coordinación. En la octava, dedicada a la extracción de parámetros físicos a partir de los espectros es donde se enmarca el presente TFM

En el presente TFM se propone localizar y modelizar el mejor algoritmo de predicción de la temperatura efectiva sobre las estrellas Enanas Ultrafrías, partiendo de un conjunto de datos simulados que deben ser similares a los datos obtenidos por la sonda GAIA

Los algoritmos que realizan la tarea de predecir los valores de una función a partir de sus variables se conocen como algoritmos de regresión.

### 2.2 Antecedentes

La novedad en la investigación del presente trabajo fin de máster es también el gran inconveniente del mismo, no existe investigación previa de la cual partir como origen.

La estrella Enana Ultrafría, es un tipo de estrella de muy reciente descubrimiento de forma que existe muy poca información referente a ella. Por lo tanto, su el análisis y procesamiento de los datos es un gran desconocido, no se dispone de muchos datos sobre ellas y se conocen unos pocos espectros "reales" sobre los cuales poder trabajar

Por este motivo, tal y como se comenta más adelante, en el apartado 2.4 Orígenes de datos, se ha partido de modelos simulados generados con Phoenix por France Allard, bajo determinadas condiciones simuladas

No existen artículos ni referencias sobre la aplicación de las técnicas de minería de datos sobre este tipo de estrella. Por lo tanto, para la ejecución del presente Trabajo Fin de Máster, se partirá de los conocimientos previos del Departamento de Inteligencia Artificial de la UNED aplicado a los conjuntos de datos astrofísicos en general.

### 2.3 Objetivos

Como ya se ha comentado, la CU8, se dedica a la extracción de parámetros físicos a partir de los espectros obtenidos por la sonda GAIA.

Los algoritmos que realizan la tarea de predecir los valores de una función a partir de sus variables se conocen como algoritmos de regresión. En el presente TFM se propone localizar y modelizar el mejor algoritmo de predicción de la temperatura efectiva sobre las estrellas Enanas Ultrafrías.

Se parte de muy poca información inicial de análisis de este tipo de estrellas, también se parte de una información, todavía hoy por hoy, imprecisa del tipo de ruido que la sonda aportará sobre los parámetros astrofísicos recogidos

Partiendo del desconocimiento inicial expuesto anteriormente, en el presente proyecto se plantean objetivos a corto, medio y largo plazo.

Los objetivos a corto plazo se centran en en preprocesado de los datos. El presente proyecto debe ser capaz de responder a las siguientes preguntas:

- Orden de magnitud de los datos

  Tipo de Normalización a aplicar sobre el conjunto de entrenamiento, teniendo en cuenta que nos encontraremos con ruido, y por lo tanto el sistema debe ser robusto al mismo.

  Estimación del ruido que la sonda GAIA aportará a los datos.

- Obtención de conjuntos de datos de validación de los algoritmos de predicción para diferentes órdenes de magnitud del ruido.

  Obtención de conjuntos de datos con ruido sobredimensionado para testar la robustez del sistema predictivo a modelizar

  Estudio de diferentes técnicas de suavizado sobre los conjuntos de datos con ruido empleados para validación y definición de la técnica que, a priori, mejor suaviza el mismo

- Estudio de reducción de la dimensionalidad del espectro. Se han introducido en el presente proyecto procedimientos de selección de atributos que deberán permitir reducir la dimensionalidad de forma que produzcan los menores efectos perturbadores en el proceso de aprendizaje y en los posteriores de clasificación

  Estudio de selección de Atributos. Los datos resultantes son extraidos desde el conjunto original formando un subconjunto del original. El gran problema de este enfoque es que es necesario resolver cuales son los más significativos respecto a las clases del conjunto de datos.

Estudio de transformación de Atributos. En este caso se probarán diferentes transformaciones de los datos de forma que los resultantes puedan ser ordenados sistemáticamente en cuanto a su relevancia

Los objetivos a medio plazo, quizás se pueden considerar como objetivos principales del presente proyecto, y se centran principalmente en:

Obtener y modelizar el mejor algoritmo de predicción de la temperatura efectiva, resolviendo los objetivos a corto plazo (preprocesado de los datos)

- Evaluar el sistema de regresión que mejor se adapte a las características de las variables regresoras y de la variable respuesta (o clase, en este caso la temperatura efectiva), teniendo en cuenta que el problema a resolver será tratar de predecir o modelar la respuesta a partir de estas variables regresoras.

Considerando el origen de los conjuntos de datos simulados, los cuales se define en el apartado 2.4, damos por supuesto que los elementos del conjunto de entrenamiento están correctamente clasificados sin error, de forma que los clasificadores a estudiar en el presente Trabajo son supervisados.

Los objetivos a largo plazo, se transforman a modo de continuidad del presente Trabajo Fin de Master, teniendo en cuenta que la sonda GAIA se lanza en el 2013, se pretende dar continuidad y validez a los sistemas y algoritmos aportados mediante el desarrollo de una Tesis Doctoral que finalizará con la validación real de los datos aportados por la sonda

### 2.4 Orígenes de Datos: Parámetros astrofísicos de referencia.

Como ya se ha comentado, la estrella Enana Ultrafría, es un tipo de estrella de muy reciente descubrimiento de forma que existe muy poca información referente a ella.

Por lo tanto, su el análisis y procesamiento de los datos es un gran desconocido, no se dispone de

muchos datos sobre ellas y se conocen unos pocos espectros "reales" sobre los cuales poder trabajar.

El único tipo de información sobre enanas ultrafrías se obtiene desde el espectro electromagnético

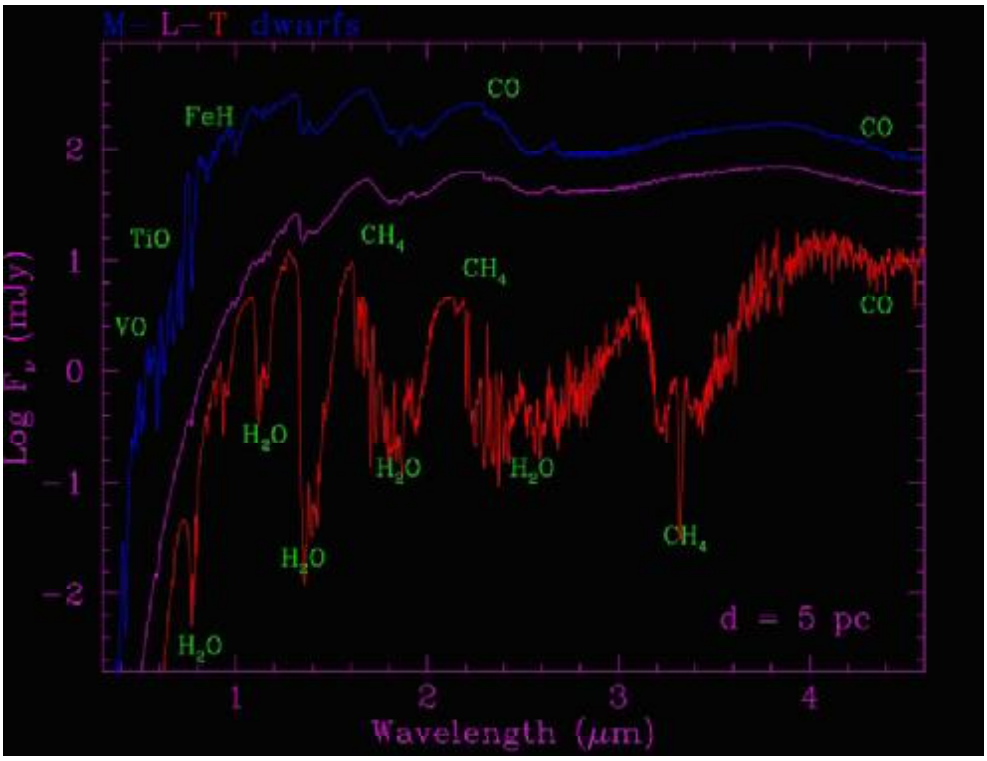


Figura 4: Espectro electromagnético de las Enanas Ultrafrías

Este modelo está basado sobre los principios físicos y las bases de datos espectroscópicas y termoquímicas las cuales se han completado lo mejor posible

Examinando las propiedades de las estrellas frías, se sabe que tienen temperaturas efectivas por debajo de 4000 K

Se distinguen entre enanas frías cuyo logaritmo de la gravedad es mayor de 3.5 (log g > 3.5) y gigantes frías cuyo logaritmo de la gravedad es menor de 3.5 (log g < 3.5)

Hay grids de modelos básicos sobre las candidatas a enanas M, subenanas M y enanas marrones partiendo de los siguientes rangos de parámetros

$$500 <= T_{eff} <= 4000\ K$$

$$3.5 <- \log g <- 5.5$$

$$-4.0 <- [M/H] <- +0.5$$

La ecuación de estado incluye 105 moléculas y más de 27 estados de ionización de 39 elementos.

En el espectro de algunas aproximaciones, se definen sobre 300 bandas moleculares de TiO, VO, CaH y FeH, incluyendo 2 millones de lineas espectrales, 42 millones de lineas atómicas y 700 millones de líneas moleculares ($H_2$, CH, NH, OH, MgH, SiH, $C_2$, CN, CO, SiO)

El presente TFM usa los modelos simulados de France Allard (http://perso.ens-lyon.fr/france.allard/) basados en el código de atmósferas del modelo Phoenix (desarrollado por el grupo de teoria del Observatorio de Hamburgo).

PHOENIX es un código estático y radial (1D) para cálculo de atmósferas estelares y planetarias actualizadas.

Obtiene fases de expansión relativistas y planetas extrasolares irradiados por una estrella con equilibrio hidrostático y aproximaciones simétricas esféricas

Puede calcular atmósferas y espectros sobre el diagrama HR incluyendo las principales secuencias de estrellas: Gigantes, enanas blancas, estrellas con viento, estrellas T Tauri, novas, supernovas, enanas marrones y planetas gigantes extrasolares. Su investigación se concentra en:

- Novae
- Supernovae
- Cool stars
- Substellar objects
- Radiation transport

- Algorithms
- Support for the GAIA mission

Los modelos son calculados asumiendo LTE (equilibrio termodinámico local), atmosferas plano-paralelas y conversión de energia y equilibrio hidrostático.

El espectro sintético de baja resolución está disponible para una estimación de temperatura de espectros observados donde la gravedad no tiene influencia significante.

A baja temperatura ($T_{eff} < 3000$ K) tiene lugar la formación de polvo

PHOENIX considera la composición quimica de la atmósfera, la formación de unas 600 especies de fases de gases, 1000 líquidos y cristales, además de una opacidad de 30 tipos de grano diferentes (e.g. $Al_2O_3$, $MgAl_2O_4$, iron, $MgSiO_3$, $Mg_2SiO_4$, carbono amorfo, SiC, algún silicato de carbono)

El código PHOENIX considera cuatro escenarios, de los cuales se emplean 2 en el presente proyecto:

- *Modelos Cond:* Los modelos COND incluyen la formación de granos de polvo pero todo el polvo se decanta en las capas profundas de la atmósfera. Por lo tanto, no contribuye a la opacidad de ésta.
- *Modelos Dusty:* Los modelos DUST incluyen el cálculo de opacidades debidas a la nube de polvo, considerada desde la base de la nube hasta la capa superior de la atmósfera. La distribución del polvo depende del tamaño de los granos, de la aceleración radiativa y de la porosidad del polvo

Sobre estos modelos atmosféricos, según la nomenclatura de France Allard se usan los modelos llamados

*AMES-Dusty*: para temperaturas efectivas (en grados Kelvin) 2500K >= Teff >= 1500K (Enanas marrones /planetas extrasolares sin irradiación, con opacidad DUST)

*AMES-Cond.* para temperaturas efectivas Teff < 1600K (Enanas marrones /planetas extrasolares sin irradiación, con opacidad COND)

A partir de estos datos, DPAC-CU2/CU8 ha simulado espectros BP (Blue Photometer) y RP (Red Photometer) de estos modelos COND y DUST para diferentes magnitudes aparentes (8, 11, 15), aunque debemos tener en cuenta que la sonda GAIA reconocerá los objetos del cielo en su totalidad hasta una magnitud 20.

Para cada modelo físico (Cond o Dusty), existen dos conjuntos de datos diferentes

NOM para los llamados grid NOMINALES con valores equiespaciados para teff y log(g) que reflejan la disponibilidad inicial de las atmósferas modelo

RAN para el grid de valores de temperatura efectiva y logaritmo de la gravedad generados aleatoriamente según las trazas evolutivas de este tipo de objetos, interpolados desde el grid nominal de France Allard.

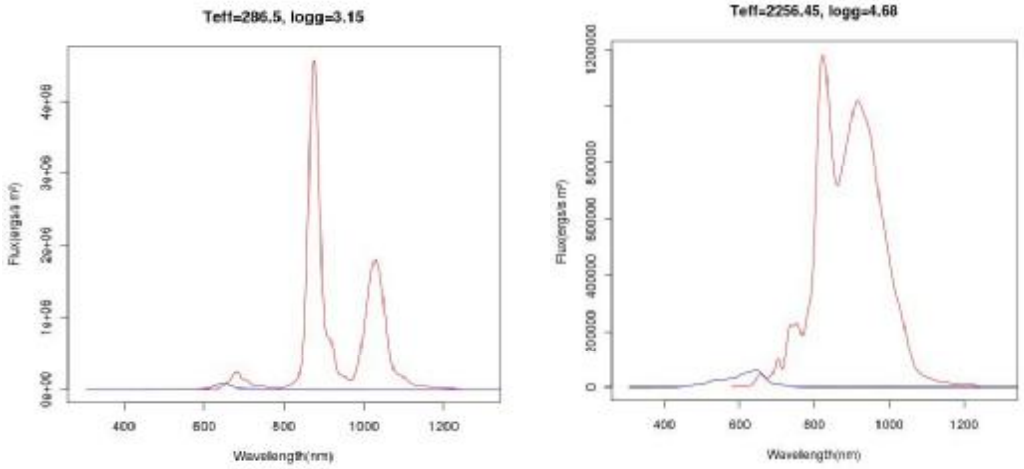


Figura 5: Espectro electromagnético de los modelos de France Allard. A la izquierda un espectro de COND, a la derecha un espectro de DUST. En azul, el espectro BP y en rojo el espectro RP según han sido simulados por la CU2 del DPAC.

Las Enanas Ultrafrías son un tipo de estrella cuyo espectro electromagnético se encuentra situado en

la franja RP, tal y como se observa en la Figura 5, el espectro en la franja BP (en azul) prácticamente es nulo en comparación con la franja RP (en rojo)

### 2.5 Conjuntos de datos, planteamiento del proyecto

#### 2.5.1 conjuntos de datos: conjuntos de entrenamiento y validación

En resumen, del apartado 2.4 se concluye que se parte de varios conjuntos de datos correspondientes a:

modelos NOM: valores nominales

modelos RAN: valores aleatorios, interpolados desde los nominales

Para cada uno de los conjuntos de datos NOM y RAN, existen dos subconjuntos correspondientes con:

submodelos COND para temperaturas entre $(100 < Teff < 1600)$.

- Submodelos DUST para un rango de temperaturas entre $(1500 < Teff < 2500)$

Cada conjunto de datos está compuesto por una serie de espectros del fotómetro rojo (RP) compuesto de 180 atributos (flujos medidos en diferentes longitudes de onda) que definen el espectro de cada candidata a Enana Ultrafría

Los espectros del fotómetro azul (BP) no serán adecuados a la tarea de regresión debido a la escasa luminosidad de las enanas ultrafrías en las longitudes de onda corta (tal y como se hace visible en la Figura 5 mostrada anteriormente).

También hay que considerar que DPAC-CU2/CU8 ha simulado espectros BP/RP de estos submodelos COND y DUST para diferentes magnitudes aparentes (G 8, 11, 15).

Sin embargo por la información que dispone el Departamento de Inteligencia Artificial de la UNED, se prevé que las enanas ultrafrías que detecte la sonda GAIA serán mucho más débiles (magnitudes aparentes entre 18 y 20).

En el *Anexo 7.1.1 Obtención de ficheros de espectros NOM, RAN, RAN COND y RAN DUST* se

aporta el código shell empleado para la obtención de los conjuntos de datos NOM (para entrenamiento) y RAN (para validación) desde los modelos aportados por DPAC-CU2/CU8.

Un código similar se ha empleado para generar los conjuntos de datos para validación COND y DUST sobre RAN

En los estudios iniciales de preprocesado, para evaluar el sistema de normalización, se emplea un único conjunto de datos para validación de los clasificadores (llamado RAN que incluye todos los espectros de magnitudes aparentes 8, 11 y 15, y subconjuntos COND y DUST para cada una ellas) obteniendo un conjunto de datos de 33,000 espectros.

En fases posteriores se generarán conjuntos de datos de validación para diferentes magnitudes, los cuales incorporarán ruido

Se deberán realizar estudios de suavizado sobre estos nuevos conjuntos de validación.

Con motivo de agilizar la sistemática de modelado y obtener más información sobre el comportamiento de los clasificadores, se emplearon únicamente los espectros RAN de magnitud aparente 15 diferenciando los conjuntos de datos de validación COND compuestos por 10.000 espectros de los conjuntos de datos para validación DUST formados por 1,000 espectros

El código shell de obtención de los correspondientes conjuntos de datos COND y DUST se encuentra descrito en *Anexo 7.1.1 Obtención de ficheros de espectros NOM, RAN, RAN COND y RAN DUST*.

Los espectros BP/RP proporcionados por DPAC-CU2/CU8, disponen de dos componentes media (RpFlux/BpFlux) y sigma (RpFluxerror/BpFluxerror). Media es el valor real de la medición mientras que sigma nos aporta la variabilidad de los datos

Para la obtención de los correspondientes valores media y sigma, en el *Anexo 9.1.2 Obtención de la información media y sigma de los espectros simulados BP/RP* se describe el código shell empleado

Dado que descartamos BpFlux, si dividimos RpFlux por su correspondiente factor de escala y multiplicando sigma (ruido) por su correspondiente valor, se generarán los diferentes conjuntos de

datos para magnitudes aparentes 18, 19, 20.

En el *Anexo 7.2.4 Creación de Espectros con Ruido para G18, G19 y G20* del presente proyecto se dispone del código R empleado para el cálculo de los diferentes conjuntos de datos de ruido empleados

Se debe considerar que las estimaciones de ruido proporcionadas por CU2 corresponden a los espectros al final de la misión (cinco años después del lanzamiento)

Por lo tanto, será necesario simular espectros RP en etapas intermedias de la misión en las que el número de observaciones de una estrella dada sea inferior al total (70 observaciones en promedio)

En dichas etapas intermedias la relación señal/ruido será muy inferior por lo que el sistema diseñado deberá ser especialmente robusto

El código R empleado para el cálculo de los conjuntos de datos correspondientes a COND y DUST para las magnitudes aparentes G20 a los dos años (etiquetado como G202Y) se muestra en el *Anexo 7.2.3 Creación de Espectros con Ruido G20 para primeros datos de la Sonda (a los dos años).*

Una estimación inicial acordada entre el Departamento de Inteligencia Artificial y el alumno para que el sistema sea robusto, es generar tres conjuntos de datos con ruido muy por encima del esperado, su obtención se describe en el *anexo 7.2.5 Creación de Espectros con Ruido sobredimensionado para robustez del sistema (X2, X5, X10)*

La obtención de la clase desde los datos proporcionados por DPAC-CU2/CU8 para cada uno de los conjuntos de datos comentados se describe en el *Anexo 7.1.3 Obtención de la Clase para NOM, RAN, COND y DUST.*

A lo largo de la ejecución del presente trabajo se han considerado diferentes conjuntos de datos de entrenamiento y validación

A partir de los espectros RAN proporcionados por DPAC-CU2/CU8 para magnitudes aparentes G8, G11 y G15, se han calculado los siguientes conjuntos de datos de 33.000 espectros cada uno:

| | RAN G8 | | RAN G11 | | RAN G15 | | |
|---|---|---|---|---|---|---|---|
| Conjunto de datos | COND | DUST | COND | DUST | COND | DUST | Nº Espectros |
| RAN G18 | X | X | X | X | X | X | 330000 |
| RAN G19 | X | X | X | X | X | X | 330000 |
| RAN G20 | X | X | X | X | X | X | 330000 |

Tabla 1: Conjunto de datos obtenidos a partir de RAN

A partir de los espectros RAN de magnitud aparente G15, se han calculado, obtenido y empleado los siguientes conjuntos de datos:

| | RAN G8 | | RAN G11 | | RAN G15 | | | espectros |
|---|---|---|---|---|---|---|---|---|
| Conjunto de datos | COND | DUST | COND | DUST | COND | DUST | Nº Espectros | Con ruido? |
| CONDRANG15 | | | | | X | | 10000 | |
| DUSTRANG15 | | | | | | X | 1000 | |
| CONDRANG20 | | | | | X | | 10000 | SI |
| DUSTRANG20 | | | | | | X | 1000 | SI |
| CONDRANG202Y | | | | | X | | 10000 | SI |
| DUSTRANG202Y | | | | | | X | 1000 | SI |
| CONDRANX2 | | | | | X | | 10000 | SI |
| DUSTRANX2 | | | | | | X | 1000 | SI |
| CONDRANX5 | | | | | X | | 10000 | SI |
| DUSTRANX5 | | | | | | X | 1000 | SI |
| CONDRANX10 | | | | | X | | 10000 | SI |
| DUSTRANX10 | | | | | | X | 1000 | SI |

Tabla 2: Conjunto de datos obtenidos a partir de RAN G15

A partir de los espectros NOM proporcionados por DPAC-CU2/CU8 , se han calculado, obtenido y empleado los siguientes conjuntos de datos

| | NOM G8 | | NOM G11 | | NOM G15 | | | espectros |
|---|---|---|---|---|---|---|---|---|
| Conjunto de datos | COND | DUST | COND | DUST | COND | DUST | Nº Espectros | Con ruido? |
| NOM | X | X | X | X | X | X | 564 | |
| NOMRUIDOX5 | X | X | X | X | X | X | 564 | SI |

Tabla 3: Conjunto de datos obtenidos a partir de NOM

Todos los conjuntos de datos mostrados en las tablas 1, 2 y 3 se han transformado y modificado en

las diferentes etapas de preprocesado, obteniendo sus conjuntos de datos equivalentes (consideremos XXX el nombre del conjunto de datos):

- XXXeuclideo
- XXXareal
- XXXPCA
- XXXKPLS
- XXXdiffusionMaps

En el caso de conjuntos de datos marcados en la columna "espectros con ruido?" en las tablas 2 y 3, además de las anteriores variantes, se han obtenido los siguientes subconjuntos de datos:

- XXXmovingaverage
- XXXwavelets

### 2.5.2 Planteamiento y ejecución del proyecto

Una vez enmarcado el proyecto, centrados los objetivos principales y descritos los diferentes conjuntos de datos a emplear, este apartado pretende definir el planteamiento a seguir en el trabajo con la finalidad de alcanzar los objetivos iniciales marcados.

El primer paso será aportar un preprocesado óptimo de los datos, para ello, las primeras pruebas irán encaminadas a la determinar la normalización a aplicar sobre los datos

Partiendo de la experiencia previa del Departamento de Inteligencia Artificial, se han considerado inicialmente emplear dos tipos diferentes de normalización: euclídea y areal

La elección final sobre cual de los dos tipos es mejor vendrá determinada por los resultados obtenidos en un conjunto de experimentos que analicen su robustez frente al ruido.

Para estos experimentos se emplearán los conjuntos de datos comentados en la tabla 1, y aplicando

la normalización euclídea o areal sobre ellos.

Estos experimentos nos permitirán descartar un tipo de normalización aplicada, o plantearnos la aplicación de nuevos tipos de normalización

Los experimentos posteriores a la elección del sistema de normalización a aplicar, vendrán marcados por la determinación del método más adecuado de suavizado del ruido, de forma que pueda eliminarse el mismo y conseguir recuperar la mayor parte de la información obtenida por la sonda GAIA.

Para estos experimentos se preverá una aproximación de la relación señal/ruido que la sonda aporta a los conjuntos de datos. Tal y como se ha descrito en el apartado anterior, se ha sobredimensionado la cantidad de ruido de forma que el sistema final obtenido sea muy robusto ante problemas de ruido no previstos (fallos en la sonda,...)

Los conjuntos de datos empleados en este conjunto de experimentos, son los marcados en la columna "espectros con ruido?" en las tabla 2, sobre los cuales previamente se habrá aplicado la normalización de datos que ofrece mejores resultados.

Para la ejecución de los experimentos de suavizado se emplearan las técnicas de Wavelets y Moving Average

La técnica de Moving Average es muy sencilla de implementar solo hay que determinar el número de vecinos a emplear, mientras que las técnicas de Wavelets representan una complejidad añadida, se dispone de numerosos filtros (Daubechies, Best Located, Least Asymetric, Haar, Coiflet,..) y diferentes órdenes

Además, la implementación de Wavelets en R dispone de varias funciones matemáticas que aplican de diferente manera las mismas.

El estudio de suavizado por lo tanto se compondrá de los siguientes experimentos: Obtención de suavizado mediante moving average de los conjuntos de datos con RUIDOX2, RUIDOX5 y

RUIDOX10 tanto para los tipos de datos COND como DUST, obtención de los diferentes suavizados mediante diferentes técnicas wavelets, filtros y órdenes

Esta fase de experimentación concluye con una comparativa entre los diferentes métodos de suavizado y la determinación del mejor modelo a aplicar sobre los conjuntos de datos iniciales

Por lo tanto sobre los conjuntos de datos comentados, recaeran las etiquetas:

- XXX-normalizacion-movingaverage
- XXX-normalizacion-wavelets-filtrousado

Para identificar las técnicas aplicadas.

La tercera fase de experimentación aborda la búsqueda del mejor algoritmo predictivo sobre los datos ya normalizados y suavizados

Para ello, una parte previa todavía de preprocesado nos permitirá conocer si la aplicación de determinados Transformadores de atributos y reductores de dimensionalidad nos aportan un beneficio en el resultado de la predicción

Este tipo de Preprocesado sobre los conjuntos de datos sumado al estudio de diferentes algoritmos de clasificación entrenados con los conjuntos de datos normalizados de la tabla 3 y validados sobre los conjuntos de datos definidos en la tabla 2, previamente normalizados y suavizados (en el caso de conjuntos con ruido) debe ser suficiente para determinar el mejor algoritmo predictivo de clasificación

Hay que comentar en la fase final del trabajo y en vista de los resultados, se incorporó un nuevo conjunto de experimentos, partiendo de un conjunto de datos de entrenamiento (referido en la tabla 3 como NOMRUIDOX5) en vista de la mejora de resultados obtenidos por otras unidades del consorcio DPAC.

## 2.6 Fundamento teórico en el Preprocesado empleado

Cuando se manipula un conjunto de datos, estos generalmente se encuentran representados en unidades de medida distintas para cada atributo y se obtiene una distancia que depende de las unidades de medida

Puede ocurrir incluso que los datos se encuentren representados en diferentes órdenes de magnitud provocando problemas de escalabilidad.

Para suavizar estos efectos, se tiende a normalizar.

En su definición más sencilla, Normalizar un vector es reducirlo a otro vector (múltiplo suyo) cuya normal es un valor concreto

### 2.6.1 Normalización Euclidea.

En el caso de la normalización Euclidea, el valor de la normal es 1.

La normalización euclidea se consigue multiplicando cada uno de los componentes del vector por

$$\frac{1}{|v|}$$

Empleando este tipo de normalización se solventa el inconveniente de los efectos de unidades de medida distintas de las variables y obtenemos un nuevo valor equivalente que no dependerá de la escala de los datos

La normalización euclidea, a pesar de su sencillez de cálculo y tiene un grave inconveniente El RUIDO.

La normalización euclidea es sensible a los valores de las variables: las diferencias entre valores de variables medidas con valores altos contribuirán en mucha mayor medida que las diferencias entre los valores de las variables con valores bajos

Como consecuencia de ello, los cambios de escala determinarán, también, cambios en la normalización entre los atributos

Una posible vía de solución de este problema es la utilización de la normalización por area

La normalización euclídea será, a priori, recomendable sólo cuando las variables sean homogéneas y estén medidas en unidades similares y/o cuando se desconozca la matriz de varianzas Sólo en conjuntos de datos donde no se prevé la existencia de ruido.

### 2.6.2 Normalización Área 1

Por lo demostrado hasta ahora, se es consciente de que el conjunto de espectros que el sistema reciba de la sonda vendrá con RUIDO. Por lo demostrado en el apartado anterior, la normalización euclídea, a priori no será buena frente a estos espectros a la entrada del sistema.

Se deberá localizar pues un sistema de normalización robusto. La propuesta que se presenta en este Trabajo Fin de Master es el uso de una normalización del espectro a area 1.

La normalización de Área1 se consigue dividiendo cada uno de los componentes del espectro por el sumatorio de cada uno de sus componentes de forma que la suma total de todos los componentes del nuevo espectro normalizado es igual a 1.

$$a_i=\frac{v_i}{\sum v_i} \qquad \sum a_i=1$$

siendo $a_i$ la nueva componente normalizada, $v_i$ el valor actual, $\min v_i$ la componente de menor valor y $\max v_i$ la componente de mayor valor

### 2.6.3 Fundamento teórico en el Suavizado de Errores

La sonda GAIA aporta un ruido a los datos que se van a obtener, de forma que se debe estudiar la forma de eliminarlo o suavizarlo de forma que afecte lo menos posible al sistema predictivo

Existen varios métodos que se han estudiado en en presente trabajo.

#### 2.6.3.1 Wavelets

La **transformada de óndula** (frecuentemente también **transformada *wavelet***) es un tipo especial de transformada de Fourier que representa una señal en términos de versiones trasladadas y dilatadas de una onda finita (denominada óndula madre).

La teoría de óndulas está relacionada con campos muy variados. Todas las transformaciones de óndulas pueden ser consideradas formas de representación en tiempo-frecuencia y, por tanto, están relacionadas con el análisis armónico.

Las transformadas de óndulas son un caso particular de filtro de respuesta finita al impulso.

Las óndulas, continuas o discretas, como cualquier función $L^2$, responden al principio de incertidumbre de Hilbert (conocido por los físicos como principio de incertidumbre de Heisenberg), el cual establece que producto de las dispersiones obtenidas en el espacio directo y en el de las frecuencias no puede ser más pequeño que una cierta constante geométrica.

En el caso de las óndulas discretas, la dispersión de los coeficientes se ha de medir de acuerdo con la norma $l^2$ (norma 2 de series numerables).

El término original francés **ondelette**, introducido por Jean Morlet y Alex Grossmann, ha sido traducido al inglés como wavelet, y también al castellano como **ondículas**, **ondeletas** u **onditas**.

Sin embargo, por su brevedad y mayor semejanza con el paradigma latino, la palabra **óndula** (un diminutivo oculto) es quizás más apropiada para este uso.

Las Transformadas de Wavelets (WT) comprenden la Transformada Wavelet Continua (CWT) y la Transformada Wavelet Discreta (DWT). Estas son dos herramientas matemáticas que permiten el análisis de señales de manera muy similar a la Transformada de Fourier.

La diferencia fundamental está en que la WT puede entregar información temporal y frecuencial en forma cuasi-simultánea, mientras que la TF sólo da una representación frecuencial.

A pesar de la existencia de problemas con la resolución en el tiempo y frecuencia (principio de incertidumbre de Heisenberg), es posible analizar cualquier señal usando un enfoque distinto mediante la WT

Esta transformada examina la señal a diferentes frecuencias con diferentes resoluciones

Como la WT es capaz de proporcionar información del tiempo y la frecuencia, ofrece buena resolución temporal y baja resolución frecuencial en eventos de altas frecuencias

Por el contrario, ofrece buena resolución frecuencial y baja resolución temporal en eventos de bajas frecuencias.

Una función $\psi \in L2(R)$ es una wavelet ortonormal si el sistema $\{\upsilon_{j,k} : j, k \in Z\}$ proporciona una base ortonormal, donde

$$\upsilon_{j,k}(x) = 2^{j/2}\,\upsilon(2^j x - k)$$

Un sistema ortogonal de H es un subconjunto de elementos no nulos del espacio ortogonales dos a dos, i.e si ei ej son dos elementos del sistema y $i = j$, entonces $\langle ei, ej \rangle = 0$.

Si además son unitarios ( $\|ei\| = 1, \forall ei$ ) decimos que el sistema es ortonormal

Diremos que el sistema ortonormal $E \subset H$, $E = \{ei\}_{i \in I}$ es completo si $E^{\perp} = \{0\}$, i.e. si $\langle x, ei \rangle = 0, \forall i \in I$ implica que $x = 0$. Un sistema ortonormal completo se llama también base hilberiana o base ortonormal de H

El sistema Haar es el filtro Wavelet más sencillo y a partir del cual se definen otros: Daubechies, Best Located, Least Asymetric, Coiflet, estudiados en el presente proyecto.

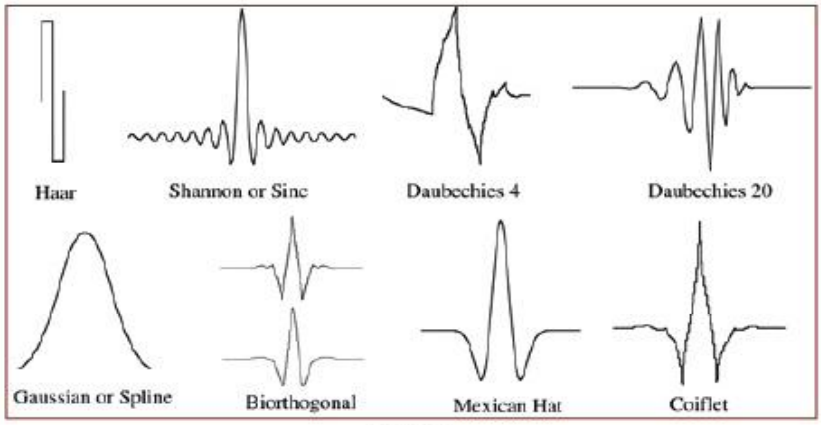


*Figure 8*

*Examples of types of wavelets*

*La Wavelet de Haar*

La wavelet de Haar está construida a partir del Análisis Multiresolución generado por la función de escala:

$$\varphi(x) = \chi_{[-1,0)}(x)$$

Como

$$\frac{1}{2}\varphi(\frac{1}{2}x) = \frac{1}{2}\chi_{[-2,0)}(x) = \frac{1}{2}\varphi(x) + \frac{1}{2}\varphi(x+1),$$

se deduce que los coeficientes del $m_0$ son

$$\alpha_0 = \tfrac{1}{2},\ \alpha_1 = \tfrac{1}{2},\ \text{y}\ \alpha_k = 0,\ \forall k \neq 0, 1$$

Así el filtro de paso bajo para la wavelet de Haar es

$$m_0(\xi) = \frac{1}{2}(1 + e^{2\pi i\xi}).$$

Calculando la transformada de Fourier de la función de escala tenemos

$$\begin{aligned}\hat{\varphi}(\xi) &= \int_{\mathbb{R}} e^{-2\pi i\xi x}\chi_{[-1,0)}(x)dx = \int_{-1}^{0} e^{-2\pi i\xi x}dx \\ &= \frac{1 - e^{2\pi i\xi}}{-2\pi i\xi} = e^{\pi i\xi}\frac{\sin \pi\xi}{\pi\xi}.\end{aligned}$$

Así, tenemos

$$\hat{\psi}(2\xi) = e^{2\pi i\xi}\overline{m_0\left(\xi+\frac{1}{2}\right)}\hat{\varphi}(xi) = e^{2\pi i\xi}\frac{\left(1-e^{-2\pi i\xi}\right)\left(e^{2\pi i\xi}-1\right)}{4\pi i\xi}.$$

Es decir,

$$\hat{\psi}(\xi) = e^{\pi i\xi}\frac{e^{\pi i\xi}-2+e^{-\pi i\xi}}{2\pi i\xi}.$$

*La construcción de Daubechies*

Cuando una wavelet tiene soporte compacto entonces el filtro de paso bajo tiene que ser un polinomio trigonométrico.

Para conservar la notación de Daubechies, se expresará el filtro de paso bajo en términos de coeficientes normalizados

$$m_0(\xi) = \frac{1}{\sqrt{2}}\sum_{k\in\mathbb{Z}} h_k e^{-2\pi ik\xi}$$

Es decir

$$\alpha_{-k} = \frac{h_k}{\sqrt{2}}.$$

Si $\varphi$ es de soporte compacto, y de clase $C^1$ entonces $m_0$ tiene que ser de la forma

$$m_0(\xi) = \left(\frac{1+e^{-2\pi i\xi}}{2}\right)^N \mathcal{L}(\xi),$$

donde $\mathcal{L}$ es un polinomio trigonométrico 1-periódico.

A un polinomio trigonométrico $m_0$ a coeficientes reales tal que

$$m_0(\xi) \neq 0 \text{ para toda } \qquad \xi \in [-1/4, 1/4]$$

le llamaremos filtro de Daubechies

Para construir $m_0$ primero determinaremos un polinomio trigonométrico $M_0$ tal que

$$M_0(\xi) = |m_0(\xi)|^2$$

Como queremos que $m_0$ sea un filtro Daubechies se tiene que cumplir que

$$M_0(\xi) = \left|\frac{e^{\pi i\xi} + e^{-\pi i\xi}}{2}\right|^{2N} L(\xi) = \cos^{2N}(\pi\xi)L(\xi)$$

siendo un polinomio trigonométrico 1-periódico tal que

$$M_0(\xi) + M_0(\xi + \frac{1}{2}) = 1,$$

$$M_0(0) = 1,$$

y donde

$$L(\xi) = \mathcal{L}(\xi)\overline{\mathcal{L}(\xi)}$$

tiene que ser un polinomio trigonométrico 1-periódico no negativo par, es decir

$$L(\xi) = Q(\cos(2\pi\xi)) = P(\sin^2(\pi\xi)),$$

con Q y P polinomios

Existe un método de factorización espectral que nos proporciona un polinomio de este tipo

$$\mathcal{L}(\xi) = \sum_{k=0}^{N-1} b_k e^{-2\pi ik\xi}, \quad b_k \in \mathbb{R}.$$

Se consiguen así las wavelets de Daubechies D2N con función de escala

$${}_N\varphi \text{ de soporte } [0, 2N-1]$$

*Cálculo de coeficientes para Daubechies de orden 4*

En el caso de N=2, tenemos que:

$$P_2(y) = \binom{1}{0} + \binom{2}{1} y = 1 + 2y,$$

$$L(\xi) = 1 + 2\sin^2(\pi\xi) = 2 - \frac{1}{2}e^{2\pi i\xi} - \frac{1}{2}e^{-2\pi i\xi}.$$

queremos determinar primero los coeficientes reales $b_0$, $b_1$ tales que

$$\left(b_0 + b_1 e^{-2\pi i\xi}\right)\left(b_0 + b_1 e^{2\pi i\xi}\right) = L(\xi) = 2 - \frac{1}{2}e^{2\pi i\xi} - \frac{1}{2}e^{-2\pi i\xi}.$$

igualando coeficientes y usando que

$$b_0 + b_1 = \mathcal{L}(0) = m_0(0) = 1$$

Obtenemos que:

$$b_0 = \frac{1+\sqrt{3}}{2} \qquad b_1 = \frac{1-\sqrt{3}}{2}.$$

Asi el filtro tiene que ser

$$\begin{aligned} m_0(\xi) &= \left(\frac{1+e^{-2\pi i\xi}}{2}\right)^2 \left(\frac{1+\sqrt{3}}{2} + \frac{1-\sqrt{3}}{2}e^{-2\pi i\xi}\right) \\ &= \frac{1}{\sqrt{2}}\sum_{k=0}^{3} h_k e^{-2\pi ik\xi}. \end{aligned}$$

Igualamos coeficientes y obtenemos

| $k$ | $h_k$ |
|---|---|
| 0 | $\frac{1}{\sqrt{2}}\frac{1+\sqrt{3}}{4} = 0{,}4829629131445341...$ |
| 1 | $\frac{1}{\sqrt{2}}\frac{3+\sqrt{3}}{4} = 0{,}8365163037378079...$ |
| 2 | $\frac{1}{\sqrt{2}}\frac{3-\sqrt{3}}{4} = 0{,}2241438680420134...$ |
| 3 | $\frac{1}{\sqrt{2}}\frac{1-\sqrt{3}}{4} = -0{,}1294095225512604...$ |

Figura 7 - Cuadro de coeficientes de la wavelets de Daubechies D4

En definitiva, el análisis de wavelets no sólo nos da las frecuencias principales, sino que nos indica cuándo ocurren y cuál es su duración

La transformada de Wavelets fue diseñada originalmente para estudiar señales **no** estacionarias.

Como presenta covariancia ante retrasos, parece ser la mejor herramienta para estudiar señales con espectro de ley de potencias

Como ya se ha comentado, se trata de un análisis de tiempo-frecuencia. Este análisis es capaz de revelar aspectos de los datos como tendencias, puntos de quiebre, discontinuidades en las derivadas, y auto-similaridad.

El análisis de wavelets puede muchas veces comprimir o eliminar ruido sin degradación apreciable. Esa será la intención de su aplicación en el presente trabajo

#### 2.6.3.2 Moving Average

Otra de las herramientas empleadas en el presente trabajo para el tratamiento de ruido es el metodo de las medias móviles o moving average.

Es un método utilizado para analizar un conjunto de datos en modo de puntos para crear series de promedios. Asi las medias móviles son una lista de números en la cual cada uno es el promedio de un subconjunto de los datos originales

Se puede calcular una serie de medias móviles para cualquier serie temporal Se usa para demanda estable, sin tendencia ni estacionalidad; suaviza las fluctuaciones de plazos cortos, resaltando asi las tendencias o ciclos de plazos largos

Existen varias posibilidades de aplicación de las moving averages:

La media móvil previa, es una **media móvil simple** (SMA) es la media aritmética de los $n$ datos anteriores. Mientras más grande sea $n$, mayor será la influencia de los datos antiguos.

La media móvil central, o también llamada media móvil exponencial, en lugar de utilizar datos anteriores, se utilizan también datos posteriores a aquél del cual se quiere obtener la media

La Media móvil ponderada, es una media multiplicada por ciertos factores, que le dan determinado peso a determinados datos.

Para su aplicación en el presente proyecto se ha optado por una media móvil central de 5 elementos dos anteriores y dos posteriores al valor del componente.

Expresada mediante formula:

$$valor_i = \frac{(valor_{i-2}) + (valor_{i-1}) + (valor_i) + (valor_{i+1}) + (valor_{i+2})}{5}$$

considerando $valor_i$, como el componente del espectro en la posición i, y los valores $valor_{i-2}$, $valor_{i-1}$, $valor_{i+1}$, $valor_{i+2}$, los valores anteriores y posteriores a la posición i.

## 2.7 Fundamento teórico en los Sistemas de Transformación de Atributos empleados: Reducción de dimensionalidad

La maldición de la dimensionalidad es una expresión acuñada por Bellman en los años 60 para referirse al hecho de que el número de patrones necesarios para estimar una cierta función con un determinado grado de precisión crece exponencialmente con el número de variables.

La complejidad de los procedimientos de aprendizaje y de los clasificadores en si mismos depende de dos factores primarios el volumen de muestras de aprendizaje y la dimensionalidad del espacio de representación.

Los costes relativos al volumen de muestras puede reducirse si podemos aplicar técnicas de condensación de los conjuntos de datos

Para reducir el coste de la dimensionalidad del espacio de representación se han introducido en el presente proyecto procedimientos de selección de atributos que permiten reducir esta dimensionalidad de forma que produzcan los menores efectos perturbadores en el proceso de aprendizaje y en los posteriores de clasificación.

Esta cuestión conecta con el problema de valoración de la representatividad de cada atributo, es decir hasta que punto es importante o accesorio, significativo o redundante.

Para formalizar el problema de la Reducción de la Dimensionalidad, intentaremos expresarlo en la siguiente forma, para cada muestra i determinar una selección o transformación de atributos:

$$x_{ij} \rightarrow y_{ik} \qquad j = 1, \ldots, n; k = 1, \ldots, l \qquad l < n$$

Tal que transforma los datos originales $\mathbf{x} \in \mathbf{R}^n$ en los de salida $\mathbf{y} \in \mathbf{R}^l$ de menor dimensionalidad donde n > l

Se emplean generalmente dos aproximaciones para afrontar este problema:

Selección de Atributos: En este caso los datos resultantes son extraídos desde el conjunto original formando un subconjunto del original. El gran problema de este enfoque es que es necesario resolver cuales son los más significativos respecto a las clases del conjunto de datos.

- Transformación de Atributos: En este caso se genera una transformación de los datos de forma que los resultantes pueden ser ordenados sistemáticamente en cuanto a su relevancia

  Existen técnicas que simplifican el problema anterior al generar conjuntos significativos en el número deseado, l, o bien conjuntos sistemáticamente ordenados en cuanto a relevancia.

Para resolver el primer enfoque es necesario disponer de la función de distribución probabilística de los datos, lo cual no siempre es posible

En el presente trabajo analizaremos la segunda opción referente a generar transformaciones de los datos que posibilitan una selección sistemática

Consideraremos varias técnicas que se utilizan frecuentemente como son el Análisis de

Componentes Principales, los Mapas de difusión y Filtro Kernel más Parcial Least Square como clasificador

Para determinar el tipo de conjunto de datos con el que se trabajara en el presente proyecto, se estudiará dos algoritmos diferentes de reducción de dimensionalidad por transformación: Un trasformador lineal (PCA) y un no lineal (DiffusionMaps).

### 2.7.1 Análisis de Componentes Principales

En estadística, el análisis de componentes principales (en español ACP, en inglés, PCA) es una técnica utilizada para reducir la dimensionalidad de un conjunto de datos. Intuitivamente la técnica sirve para hallar las causas de la variabilidad de un conjunto de datos y ordenarlas por importancia

Técnicamente, el PCA busca la proyección según la cual los datos queden mejor representados en términos de mínimos cuadrados. El PCA se emplea sobre todo en análisis exploratorio de datos y para construir modelos predictivos. El PCA comporta el cálculo de la descomposición en autovalores de la matriz de covarianza, normalmente tras centrar los datos en la media de cada atributo

El PCA construye una transformación lineal que escoge un nuevo sistema de coordenadas para el conjunto original de datos en el cual la varianza de mayor tamaño del conjunto de datos es capturada en el primer eje (llamado el Primer Componente Principal), la segunda varianza más grande es el segundo eje, y así sucesivamente. Para construir esta transformación lineal debe construirse primero la matriz de covarianza o matriz de coeficientes de correlación

Debido a la simetría de esta matriz existe una base completa de vectores propios de la misma. La transformación que lleva de las antiguas coordenadas a las coordenadas de la nueva base es precisamente la transformación lineal necesaria para reducir la dimensionalidad de datos. Además las coordenadas en la nueva base dan la composición en factores subyacentes de los datos iniciales.

Una de las ventajas del PCA para reducir la dimensionalidad de un grupo de datos, es que retiene

aquellas características del conjunto de datos que contribuyen más a su varianza, manteniendo un orden de bajo nivel de los componentes principales e ignorando los de alto nivel. El objetivo es que esos componentes de bajo orden a veces contienen el "más importante" aspecto de esa información

Supongamos que existe una muestra con $n$ individuos para cada uno de los cuales se han medido $m$ variables $F_i$ (aleatorias) . El ACP permite encontrar un número de factores subyacentes $p < m$ que explican aproximadamente el valor de las $m$ variables para cada individuo. El hecho de que existan estos $p$ factores subyacentes puede interpretarse como una reducción de la dimensionalidad de los datos: donde antes necesitabamos $m$ valores para caracterizar a cada individuo ahora nos bastan $p$ valores. Cada uno de los $p$ encontrados se llama **componente principal**, de ahí el nombre del método.

Existen dos formas básicas de aplicar el ACP:

1. Método basado en la matriz de correlación, cuando los datos no son dimensionalmente homogéneos o el orden de magnitud de las variables aleatorias medidas no es el mismo.
2. Método basado en la matriz de covarianzas, que se usa cuando los datos son dimensionalmente homogéneos y presentan valores medios similares.

El método parte de la matriz de correlaciones, consideremos el valor de cada una de las $m$ variables aleatorias $F_j$ Para cada uno de los $n$ individuos tomemos el valor de estas variables y escribamos el conjunto de datos en forma de matriz

$$(F_j^\beta)_{j=1,...,m}^{\beta=1,...,n}$$

Obsérvese que cada conjunto

$$\mathcal{M}_j = \{F_j^\beta | \beta = 1, ..., n\}$$

puede considerarse una muestra aleatoria para la variable $F_j$ A partir de los m x n datos correspondientes a las $m$ variables aleatorias, puede construirse la matriz de correlación muestral,

que viene definida por:

$$\mathbf{R} = [r_{ij}] \in M_{m\times m} \qquad \text{donde} \qquad r_{ij} = \frac{\mathbf{cov}(F_i, F_j)}{\sqrt{\mathbf{var}(F_i)\mathbf{var}(F_j)}}$$

Puesto que la matriz de correlaciones es simétrica entonces resulta diagonalizable y sus valores propios [redacted] verifican:

$$\sum_{i=1}^{m} \lambda_i = 1$$

Debido a la propiedad anterior estos $m$ valores propios reciben el nombre de pesos de cada uno de los $m$ componentes principales. Los **factores principales** identificados matemáticamente se representan por la base de vectores propios de la matriz R. Está claro que cada una de las variables puede ser expresada como combinación lineal de los vectores propios o componentes principales

En el Método basado en las covarianzas, el objetivo es transformar un conjunto dado de datos **X** de dimensión $n \times m$ a otro conjunto de datos **Y** de menor dimensión $n \times l$ con la menor perdida de información útil posible utilizando para ello la matriz de covarianza.

Se parte de un conjunto $n$ de muestras cada una de las cuales tiene $m$ variables que las describen y el objetivo es que, cada una de esas muestras, se describa con solo $l$ variables, donde $l < m$. Además, el número de componentes principales $l$ tiene que ser inferior a la menor de las dimensiones de X

[redacted]

Los datos para el análisis tienen que estar centrados a media 0 (restándoles la media de cada columna) y/o autoescalados(centrados a media 0 y dividiendo cada columna por su desviación estándar).

$$\mathbf{X} = \sum_{a=1}^{l} \mathbf{t}_a \mathbf{p}_a^T + \mathbf{E}$$

Los vectores $t_a$ son conocidos como *scores* y contienen la información de cómo las muestras están

relacionadas unas con otras además, tienen la propiedad de ser ortogonales. Los vectores $P_x$ se llaman *loadings* e informan de la relación existente entre las variables y tienen la cualidad de ser ortonormales. Al coger menos componentes principales que variables y debido al error de ajuste del modelo con los datos, se produce un error que se acumula en la matriz E.

El PCA se basa en la descomposición en vectores propios de la matriz de covarianza. La cual se calcula con la siguiente ecuación:

$$\mathrm{cov}(X) = \frac{X^T X}{n-1}$$

$$\mathrm{cov}(X)\, P_x = \lambda_x P_x$$

$$\sum_{x=1}^{n} \lambda_x = 1$$

Donde $\lambda_x$ es el valor propio asociado al vector propio $P_x$. Por último,

$$t_x = X P_x$$

Esta ecuación la podemos entender como que $t_x$ son las proyecciones de X en $P_x$, donde los valores propios $\lambda_x$ miden la cantidad de varianza capturada, es decir, la información que representan cada uno de los componentes principales. La cantidad de información que captura cada componente principal va disminuyendo según su número es decir, el componente principal número uno representa más información que el dos y así sucesivamente. La aplicación del PCA está limitada por varias asunciones:

- Asunción de linealidad: Se asume que los datos observados son combinación lineal de una cierta base.
- Importancia estadística de la media y la covarianza: el PCA utiliza los vectores propios de la matriz de covarianzas y sólo encuentra las direcciones de ejes en el espacio de variables considerando que los datos se distribuyen de manera gaussiana

Se puede demostrar como PCA es incapaz de tratar conjuntos de datos no-lineales. En la figura 10, la imagen de la izquierda muestra una distribución en espiral en las dos primeras componentes principales. En la imagen de la derecha (proyección en el eigenvalor de mayor variabilidad).

El coloreado muestra la dependencia en los componentes $x_1$ y $x_2$ mediante la función:

$$f(t) = (t\cos(t), t\sin(t))$$

Se puede observar una solapación de los puntos azules, amarillos y rojos en la línea central proyectada, se observa como la información más pequeña se pierde. Es decir, la información más geométrica se pierde al proyectar los datos sobre el eigenvalor de mayor variabilidad

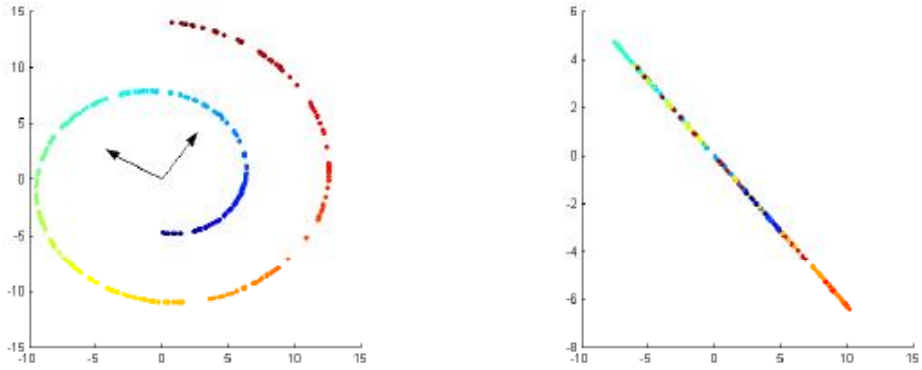

Figura 8: Perdida de información usando PCA

### 2.7.2 Diffusion Maps

Diffusion maps es una técnica para reducción de dimensionalidad que suele emplearse cuando los samples del conjunto de datos no se encuentran uniformemente distribuidos.

Coifman y Lafon proveen una nueva nnotización para normalizar gráficos Laplacianos empleando las distancias de difusión

Estas distancias proveen diferentes geometrias multiescala dependiendo de cómo la matriz de camino aleatorio es iterada

Usando el mismo camino aleatorio de los gráficos Laplacianos:

Laplacian Random Walk

$$L_{rw} = D^{-1}L = I - D^{-1}W$$

teniendo encuenta:

$\lambda$ es un eigenvalor de $L_{rw}$ con eigenvector v si y solo si $\lambda$ y v resuelven el eigenproblema generalizado

$$Lv = \lambda Dv$$

$L_{rw}$ es positivo y semidinido con el primer $\lambda_1 = 0$ eigenvalor de y la constante del vector1 correspondiente al eigenvector

$$0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$$

Todos los eigenvectores son reales y cumplen

Aplicando una pequeña diferencia, antes de la definición de $L_{rw}$, llamaremos P al operador de difusion tal que:

$$P = D^{-1}W$$

donde cada entrada P viene definida por:

$$p_{ij} = k(x_i, x_j)/d(i)$$

vista como el kernel de transición de la Cadena de Markov en G. En otras palabras $p_{ij}$ define la probabilidad de transición de ir del estado i al estado j en un paso. Por eso P define la cadena de Markov completa y $P^t$ nos dá la probabilidad de transición de cada punto a otro en t pasos temporales

Los mapas de difusión pueden calcularse usando eigenvectores $\varphi$ y eigenvalores $\lambda$ de P

$$D_t(x, y) = \left( \sum_{l \geq 1} \lambda_l^{2t} (\psi_l(x) - \psi_l(y))^2 \right)^{\frac{1}{2}}.$$

Usando el teorema espectral del espacio de Hilbert, y de echo las eigenfunciones de P son ortonormales

Explotando esto

$$1 = \lambda_0 > |\lambda_1| \geq |\lambda_2| \geq \ldots$$

La distancia de difusión pueden ser aproximadas usando los s primeros eingenvalores (s dimensiones).

### 2.8 Fundamento teórico en los Clasificadores de Regresión empleados

El problema de la regresión ha suscitado siempre mucho interés. Se pueden distinguir distintos tipos de regresión en función de las características de las variables regresoras y de la variable respuesta, pero el problema siempre es el mismo, se trata de predecir o modelar la respuesta a partir de las variables regresoras.

De este modo la regresión se aplica en multitud de áreas, desde la economía a la meteorología, e incluso hay autores que prefieren ver la clasificación supervisada como una regresión en la que la variable dependiente es la clase

Existen numerosas vías de investigación referentes a predicción de series temporales. Una de las grandes áreas del Análisis de Datos Funcionales (FDA) es la clasificación.

Independientemente de las divisiones entre métodos paramétricos y no-paramétricos, los distintos tipos de algoritmos o aproximaciones, la clasificación engloba dos grandes ramas conocidas como clasificación supervisada y clasificación no supervisada.

El principal objetivo de la clasificación no supervisada también llamada clustering, es dividir un conjunto de datos (típicamente grande) $x_1,\ldots x_n$ en un cierto número de clases k definido, de forma que los miembros de cada clase o cluster tengan algún tipo de similitud. Esta similitud vendrá definida por el algoritmo y la métrica utilizados

Uno de los principales retos en clasificación no supervisada es la elección del número de clases, aunque algunos algoritmos aportan sugerencias para esta elección. El clustering se suele utilizar cuando se sospecha de una división oculta en subgrupos de un conjunto de datos y está encaminado

a revelar dicha estructura y facilitar la comprensión del problema

Por su parte, la clasificación supervisada se aplica a problemas en los que k poblaciones (o clases) P1,. ..Pk vienen dadas y bien definidas. En este marco, los individuos son variables aleatorias X, y el conjunto de datos disponible es $\{(X_i, Y_i), 1 \leq i \leq n\}$ el conjunto de entrenamiento. Éste se define como el conjunto de duplas donde los X.'s son observaciones independientes de X, y donde Yi es la etiqueta que indica la población a la que pertenece la i-ésima observación.

El término supervisado hace referencia a que se supone que los elementos del conjunto de entrenamiento están clasificados sin error mediante algún procedimiento no estadístico. Para que el problema tenga solución se asume que la distribución de X en cada población (X / Y = j) es distinta.

El objetivo del clasificador será, una vez entrenado, asignar una nueva observación de X a la población adecuada prediciendo Y.

Tanto en un caso como en otro existe abundante literatura y los principales programas tienen paquetes estándar para trabajar con los algoritmos más comunes.

Considerando el origen de los datos definido en el apartado 2.4, damos por supuesto que los elementos del conjunto de entrenamiento están correctamente clasificados sin error, de forma que los clasificadores a emplear en el presente Trabajo son supervisados.

A continuación se expone la definición formal de clasificador Supervisado.

Sea

$$D_n = \{(\chi_i, Y_i), 1 \leq i \leq n\}$$

el conjunto de entrenamiento con observaciones independientes que siguen la distribución de la variable aleatoria

$$\chi \in \mathcal{F}(\sim L^2[a,b]) \text{ con } Y \in \{0,1\}$$

Se define un clasificador g como la aplicación

$$g : \chi \longrightarrow \{0, 1\}$$

construida para minimizar el error de clasificación

$$P(g(\chi_i) \neq Y_i)$$

para todo i

La conocida como Regla de Bayes define el clasificador del mismo nombre del siguiente modo:

$$g^*(\chi) = I_{\{\eta(\chi)>1/2\}}$$

donde

$$\eta(x) = E(Y|X = x).$$

Es decir, la esperanza de que la observación pertenezca a una cierta clase dada la observación, e I es la función indicatriz. Este clasificador es óptimo en el sentido de que minimiza el error de clasificación definido anteriormente y el objetivo de todos los clasificadores que se construyan será acercarse lo más posible al de Bayes.

El problema de $g^*$ es que sólo se puede definir en contadas ocasiones por la dificultad de dar una expresión para $\eta$.

Por tanto, los clasificadores que se construyen tendrán siempre como cota de error, el error Bayes

Estos clasificadores

$$g_n(x) = g_n(x; D_n)$$

se construyen en la práctica a partir de la información del conjunto de entrenamiento. El error condicionado de estos clasificadores es

$$L_n = P(g_n(\chi) \neq Y|D_n)$$

y debe tender al óptimo

$$L^* = P(g^*(\chi) \neq Y)$$

De hecho, para una secuencia de clasificadores $g_n$ se mide la consistencia en función de su convergencia al error Bayes.

Así $L_n \longrightarrow L^*$ en probabilidad, o lo que es lo mismo,

$$E(L_n) \overset{n\to\infty}{\longrightarrow} L^*$$

se dice que $g_n$ es débilmente consistente

Del mismo modo, si converge casi seguramente en lugar de en probabilidad, se dice que la secuencia es fuertemente consistente

Hay múltiples formas de crear clasificadores, dos de las más comunes son:

Los llamados métodos de plug-in que sustituyen $\eta$ y aplican el clasificador de Bayes con esta nueva $\eta$. De nuevo, al desconocer la distribución conjunta de (X,Y) no se puede calcular el error de g, teniendo que recurrir a un estimador llamado riesgo empírico y definido del siguiente modo:

$$\widehat{L}_n = \widehat{L}_n(g) = \frac{1}{n}\sum_{i=1}^{n} I_{\{g(X_i)\neq Y_i\}}$$

- A partir de esta expresión se pueden definir los clasificadores basados en riesgo. La idea es elegir una clase C de clasificadores (con una estructura simple o cualquier otra propiedad interesante) y elegir como clasificador al que solucione el problema de minimización del riesgo empírico:

$$g_n^* = argmin_{g\in C}\widehat{L}_n(g).$$

Los clasificadores así construidos suelen dar mejores resultados que los plug-in. Esto puede ser debido a que no tienen que estimar $\eta$, que ya es por sí solo un problema complicado

### 2.8.1 K- Vecinos Cercanos (k-NN)

El método de los vecinos más próximos es válido tanto para datos funcionales. Sea F un espacio métrico, se clasifica X en función de los k ejemplos de entrenamiento más próximos según la métrica de F. Los empates se pueden decidir aleatóriamente o con algún otro procedimiento, y k juega en este método el papel de parámetro de suavizado.

Se puede considerar como un método plug-in tomando

$$\eta(\chi) = \eta_n(\chi) = \frac{1}{k}\sum_{i=1}^{n} I_{\{\chi_i \in k(\chi)\}} Y_i$$

y entonces

$$g_n(\chi) = I\{\eta_n(\chi) > 1/2\}$$

donde

$$\chi_i \in k(\chi)$$

hace referencia a si la i-ésima observación pertenece a los k vecinos más próximos a X. La selección del parámetro k se suele hacer minimizando el error por validación cruzada

La única diferencia entre funcional y multivariado es la elección de una distancia de forma adecuada, la cual viene definida por el espacio funcional y usualmente se toman $L^2[a,b]$ con la métrica usual o $C[a,b]$ con la métrica del supremo, si bien otras aproximaciones pueden ser interesantes

K-NN es un método ámpliamente utilizado y existen numerosas referencias a las propiedades de consistencia del método

Ya sea por los buenos resultados globales, sus buenas propiedades matemáticas o su sencillez, k-NN es un método muy popular y utilizado, y sobre el que se sigue investigando.

### 2.8.2 Estimadores basados en nucleo (Kernel Reproductor)

La teoria de kernel reproductores surge del análisis matemático para pasar después a la estadistica Reproducing Kernel es una teoria de transformación que asocia una función kernel definida positiva con un espacio de Hilbert.

Como otras teorias, los kernel reproductores se fundamentan en que problemas complejos en un cierto espacio pueden ser fácilmente resolubles en otro, y la solución óptima en éste suele serlo también en el primero

Estos sistemas empezaron a cobran importancia con los trabajos de Parzen en los años 50, dejando de ser una herramienta oscura, y se han ido estudiando y perfeccionando hasta irrumpir con fuerza en el mundo de la clasificación de patrones de las manos de las Máquinas de Vectores Soporte o los Procesos Gausianos

La función de estos métodos es simple, la idea es utilizar una función kernel para llevar un conjunto de observaciones a un espacio de Hilbert donde la distancia entre las observaciones venga determinada por el kernel. Asignando esta distancia se pueden utilizar las herramientas tradicionales.

Centrándonos en el problema de clasificación y formalizando estas ideas se define

$$K : \mathcal{F}x\mathcal{F} \longrightarrow \mathbb{R}$$

definida positiva como un Kernel Reproductor de un espacio de Hilbert $(H,\langle .,. \rangle)$ de funciones reales definidas en F que cumple:

1. Para todo $x \in \mathcal{F}$, $K(.,x) \in H$.
2. Para todo $x \in \mathcal{F}, \varphi \in H$, $\langle \varphi, K(.x) \rangle = \varphi(x)$.

Los espacios de Hilbert en los que se da esto reciben el nombre de RKHS (Reproducing Kernel Hilbert Space).

Estas propiedades hacen que podemos operar con elementos en espacios de Hilbert mediante productos escaleres en los espacios de partida y así, como se ha comentado antes, utilizar el Kernel $K(x_i,x_j)$ para meter los puntos $x_1,\ldots,x_n$ en $H$ con $K(x_i,x_j)$ como distancia entre $x_i,x_j$ y utilizar otros métodos.

Como clasificadores se estima una (RKHS asociada al kernel K) minimizando el riesgo empirico regularizado

$$\frac{1}{n}\sum_{i=1}^{n} C(\chi_i, Y_i, \hat{\eta}(\chi_i)) + J(\hat{\eta})$$

donde C es una función de coste convexa respecto al tercer argumento y $J(\cdot)$ es un término de penalización. Una elección tipica es tomar

$$C(\chi_i, Y_i, \hat{\eta}(\chi_i)) = (Y_i - \hat{\eta}(\chi_i))^2 \text{ y } J(\hat{\eta}) = \lambda \parallel \hat{\eta} \parallel_K^2$$

para un cierto parámetro de regularización $\lambda > 0$. En nuestro caso también puede definirse la función de coste como

$$C(\chi_i, Y_i, \hat{\eta}(\chi_i)) = -Y \log \hat{\eta}(\chi_i) + \log(\tfrac{1}{1-\hat{\eta}(\chi_i)}$$

Con esta función, la solución al problema de minimización es:

$$\hat{\eta}(x) = \sum_{i=1}^{n} \alpha_i K(x, \chi_i),$$

con los coeficientes reales

*Función Kernel*

La manera más simple de realizar la separación es mediante una linea recta, un plano recto o un hiperplano N-dimensional

Debido a las limitaciones computacionales de las máquinas de aprendizaje lineal, éstas no pueden ser utilizadas en la mayoría de las aplicaciones del mundo real

La representación por medio de funciones Kernel ofrece una solución a este problema, proyectando la información a un espacio de características de mayor dimensión el cual aumenta la capacidad computacional de la máquinas de aprendizaje lineal. Es decir, mapearemos el espacio de entradas X a un nuevo espacio de características de mayor dimensionalidad (Hilbert)

$$F = \{\varphi(x) \mid x \in X\} \quad x = \{x1, x2, \cdots, xn\} \quad \cdot \varphi(x) = \{\varphi(x)1, \varphi(x)2, \cdots, \varphi(x)n\}$$

*Tipos de funciones Kernel (Núcleo)*

- Polinomial-homogénea: $K(x_i, x_j) = (x_i \cdot x_j)^n$

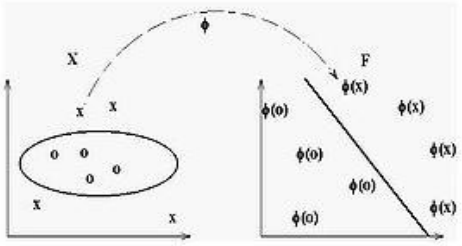


Figura 9: Kernel Polinómico

- Perceptron: $K(x_i, x_j) = \| x_i - x_j \|$

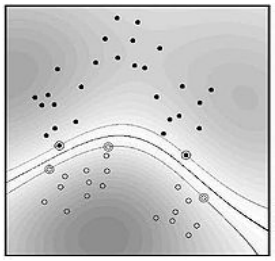

Figura 10: Kernel Perceptron

- Función de Base Radial Gaussiana separado por un hiperplano en el espacio transformado

$$K(x_i, x_j) = \exp(-(x_i - x_j)^2 / 2(\text{sigma})^2)$$

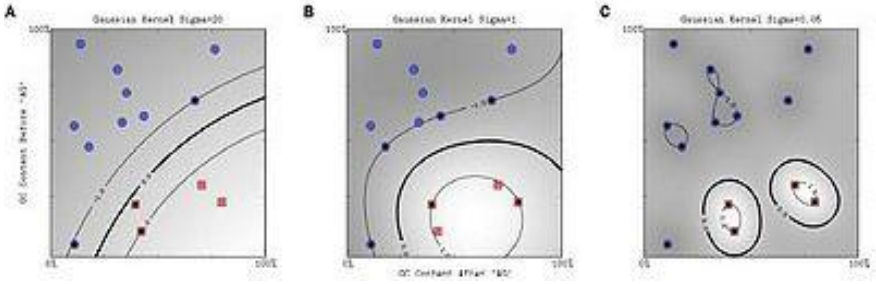


Figura 11: Kernel Base Radial Gaussiana

- Sigmoid: $K(x_i, x_j) = \tanh(x_i \cdot x_j + 0)$

### 2.8.2.1 Máquinas de Vectores Soporte (SVM)

RKHS es más una metodología general que un método definido, incluso elegidos el kernel y C aún hay que ajustar distintos parámetros. El más conocido de los procedimientos que aplica la filosofía de RKHS son las máquinas de vectores soporte

Los SVM fueron introducidos por Vapnik que fijó el marco teórico en el contexto multivariado Estos primeros trabajos se centraron en el caso binario, que sigue siendo el más representativo aunque han proliferado los trabajos, generalizaciones y modificaciones del método hasta convertirse en un referente. En el caso binario, donde las observaciones se denotan por

$$(x_i, y_i) \text{ con } x_i \in \mathbb{R}^d \text{ e } y_i \in \{-1, 1\}$$

se trata de obtener un clasificador de la forma

$$\phi_n(x) = \text{Signo}\{\langle w, \psi(x)\rangle_H + b\}, \text{ donde } \psi : \mathbb{R}^d \longrightarrow H$$

es una extensión a un espacio de Hilbert

La idea es que en el espacio de Hilbert el problema se separará más fácilmente Para ajustar los coeficientes w y b se resuelve el problema de maximización del margen entre las observaciones de

las distintas clases y la frontera de decisión.

En un primer momento no se permitían muestras mal clasificadas, pero esto era muy poco flexible y no se ajustaba a la práctica, por lo que después se permitieron, quedando el problema de maximización del siguiente modo:

$$\min_{w,b,\epsilon} \| w \|_H^2 + C \sum_{i=1}^{n} \epsilon_i$$

tal que

$$y_i(\langle w, \psi(x) \rangle_H + b) \geq 1 - \epsilon_i \text{ y } \epsilon_i \geq 0, 1 \leq i \leq n.$$

Este es un problema complicado, pero tiene una formulación dual en función de los productos internos

$$\langle \psi(x_i), \psi(x_j) \rangle_H$$

que mediante la utilización de los kernel reproductores hace innecesario conocer el espacio H y el producto interno de forma explícita, ya que elegido el kernel, existe un H y un $\psi$ con las características comentadas tales que

$$\langle \psi(x_i), \psi(x_j) \rangle_H = K(x_i, x_j).$$

Este "truco" radica la potencia del método y es la base de su éxito, además ya hay resultados de consecuencia universal bajo ciertas hipótesis. La extensión funcional surge de forma natural

Todo el mecanismo se puede adaptar a datos en un espacio funcional $\Gamma$, en lugar de $\mathrm{R}^{\mathrm{d}}$, pasando a las versiones continuas (sumario por integral, por ejemplo), sin embargo, la naturaleza de los datos funcionales hacen que no sea aplicable en muchos casos esta adaptación trivial (o no consiga buenos resultados) e invalida los resultados de consistencia

En primer lugar hay que tener en cuenta el preprocesado de datos mediante discretizaciónes, proyecciones a una base finita de funciones o cálculo de derivadas.

En cuanto a los kernel, la mayoría de los clásicos se adaptan al caso funcional, como el gaussiano

$$K(f,g) = e^{-\sigma\|f-g\|^2},$$

o el polinómico

$$K(f,g) = (1 + \langle f,g\rangle)^D.$$

sin embargo, la naturaleza funcional de los datos puede utilizarse para definir nuevos kernels, por ejemplo, combinando los clásicos con una función de mapeo.

A nivel práctico, los datos nunca son funciones perfectas sino un vector de los valores en una partición, por lo que a menudo, se toma la solución simple de aplicar el modelo de SVM estándar

Dado un conjunto de puntos, subconjunto de un conjunto mayor (espacio), en el que cada uno de ellos pertenece a una de dos posibles categorías, un algoritmo basado en SVM construye un modelo capaz de predecir si un punto nuevo (cuya categoría desconocemos) pertenece a una categoría o a la otra.

Como en la mayoría de los métodos de clasificación supervisada, los datos de entrada (los puntos) son vistos como un vector $p$-dimensional (una lista de $p$ números).

La SVM busca un hiperplano que separe de forma óptima a los puntos de una clase de la de otra, que eventualmente han podido ser previamente proyectados a un espacio de dimensionalidad superior.

En ese concepto de "separación óptima" es donde reside la característica fundamental de las SVM: este tipo de algoritmos buscan el hiperplano que tenga la máxima distancia (margen) con los puntos que estén más cerca de él mismo

Por eso también a veces se les conoce a las SVM como *clasificadores de margen máximo*. De esta forma, los puntos del vector que se etiquetados con una categoría estarán a un lado del hiperplano y los casos que se encuentren en la otra categoría estarán al otro lado.

Los algoritmos SVM pertenecen a la familia de los clasificadores lineales. También pueden ser considerados un caso especial de la regularización de Tikhonov.

En la literatura de los SVMs, se llama *atributo* a la variable predictora y *característica* a un atributo transformado que es usado para definir el hiperplano. La elección de la representación más adecuada del universo estudiado, se realiza mediante un proceso denominado selección de características. Al vector formado por los puntos más cercanos al hiperplano se le llama vector de soporte.

*Soft margin. Errores de entrenamiento*

Idealmente, el modelo basado en SVM debería producir un hiperplano que separe completamente los datos del universo estudiado en dos categorías. Sin embargo, una separación perfecta no siempre es posible y, si lo es, el resultado del modelo no puede ser generalizado para otros datos. Esto se conoce como sobreajuste (overfitting).

Con el fin de permitir cierta flexibilidad, los SVM manejan un parámetro C que controla la compensación entre errores de entrenamiento y los márgenes rígidos, creando así un margen blando (soft margin) que permita algunos errores en la clasificación a la vez que los penaliza.

#### 2.8.2.2 Procesos Gausianos

Como ya se ha comentado RKHS es más una metodología general que un método definido, incluso elegidos el kernel y C aún hay que ajustar distintos parámetros. El más conocido de los procedimientos que aplica la filosofía de RKHS son las máquinas de vectores soporte. Otra metodología son los Procesos Gausianos.

Considérese un modelo definido en términos de una combinación lineal de M funciones base dadas por los elementos del vector $\varphi(x)$ de forma tal que

$$y(\mathbf{x}) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$$

donde x es el vector de entrada y w es un vector de pesos de dimensión M . En la práctica, se desea evaluar la función en un conjunto finito de valores de x , por ejemplo en los puntos de entrenamiento $x_1 ... , x_N$ .

Se desea modelar la probabilidad conjunta $y (x_1 )... , y (x_N )$ , que se denota por el vector y con elementos $y_n = y (x_n )$ para $n = 1,... , N$ . De la ecuación, este vector está dado por

$$y = \Phi w$$

donde Φ es una matriz de diseño con elementos $\Phi_{nk} = \phi_k (x_n )$ .

Asumiendo que $p(w) = N(w | 0, \alpha^{-1} I )$ , la función de distribución sobre y sigue igualmente una distribución gaussiana $p(y) = N(y | 0, K )$ , donde K está dada como

$$\mathbf{K} = \frac{1}{\alpha}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}$$

La función de probabilidad anterior p (y) es un caso particular de un proceso gaussiano. En forma general, un proceso gaussiano se define como una distribución de probabilidad sobre funciones y (x) tal que el conjunto de valores de y (x) evaluados en un conjunto arbitrario de puntos $x_1 ,... , x_N$ tienen conjuntamente una función de distribución gaussiana.

Un punto importante de los procesos estocásticos gaussianos es que la distribución conjunta sobre N variables $y_1 ,... , y_N$ se especifica completamente con estadística de segundo orden. Para muchas aplicaciones se asume que la media del proceso es igual a cero y la matriz de covarianza se especifica evaluando y (x) para dos valores de x , lo cual permite obtener

$$E[ y (x_n ) y (x_m )] = k (x_n , x_m )$$

donde k (x, x ') se conoce como la función kernel

Una condición necesaria y suficiente para que la función k (x, x ') sea un kernel válido es que la

matriz de Gram K , cuyos elementos están dados por $k(x_n, x_m)$ , debe ser semidefinida positiva para todos los valores posibles de $\{x_n\}$ .

$$\sigma(a) = \frac{1}{1+\exp(-a)}$$

Se obtiene entonces un proceso estocástico no gaussiano sobre las funciones y (x) con y ∈ {0,1} .

Formalmente, sea $x_1, \ldots, x_N$ el conjunto de vectores de entrenamiento con sus correspondientes etiquetas de clase $t = (t_1, \ldots, t_N)$ Sea $x_{N+1}$ un vector de prueba con su correspondiente etiqueta $t_{N+1}$ . El objetivo es determinar la función de distribución predictiva $p(t_{N+1} \mid t)$ Para lograrlo, se introduce un proceso gaussiano sobre el vector $a_{N+1}$ que tiene componentes

$a(x_1), \ldots a(x_{N+1})$ A su turno, esto define un proceso no gaussiano sobre $t_{N+1}$ y, condicionándolo sobre el conjunto de entrenamiento $t_N$ , se obtiene la distribución predictiva que se requiere. El proceso gaussiano para $a_{N+1}$ toma la forma

$$p(\mathbf{a}_{N+1}) = \mathrm{N}(\mathbf{a}_{N+1} \mid \mathbf{0}, \mathbf{C}_{N+1})$$

donde la matriz de covarianza tiene elementos dados por

$$\mathbf{C}(\mathbf{x}_n, \mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m) + \upsilon\delta_{nm}$$

siendo $k(x_n, x_m)$ una función kernel semidefinida positiva y la constante υ se introduce para asegurar que la matriz C sea definida positiva. Una función kernel ampliamente utilizada es

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp\left\{-\frac{\theta_1}{2}\|\mathbf{x}_n - \mathbf{x}_m\|^2\right\} + \theta_2 + \theta_3 \mathbf{x}_n^{\mathrm{T}} \mathbf{x}_m \qquad (1.2)$$

Para el problema bielase es suficiente con predecir $p(t_{N+1} = 1 \mid t_N)$ , dado que el valor de $p(t_{N+1} = 0 \mid t_N)$ se obtiene como $1 - p(t_{N+1} = 1 \mid t_N)$ La distribución predictiva requerida está dada por

$$p(t_{N+1} = 1 \mid \mathbf{t}_N) = \int p(t_{N+1} = 1 \mid a_{N+1}) p(a_{N+1} \mid \mathbf{t}_N) \mathrm{d}a_{N+1}$$

donde $p(t_{N+1} = 1 \mid a_{N+1}) = \sigma(a_{N+1})$ .

Esta integral es intratable y debe emplearse algún método para aproximarla. En este trabajo, se emplea el método de Laplace. La idea es encontrar una aproximación gaussiana a la distribución posterior sobre $a_{N+1}$, que usando el teorema de Bayes, está dada por.

$$\begin{aligned} p(a_{N+1} \mid \mathbf{t}_N) &= \int p(a_{N+1}, \mathbf{a}_N \mid \mathbf{t}_N) \mathrm{d}\mathbf{a}_N \\ &= \frac{1}{p(\mathbf{t}_N)} \int p(a_{N+1}, \mathbf{a}_N) p(\mathbf{t}_N \mid a_{N+1}, \mathbf{a}_N) \mathrm{d}\mathbf{a}_N \\ &= \frac{1}{p(\mathbf{t}_N)} \int p(a_{N+1} \mid \mathbf{a}_N) p(\mathbf{a}_N) p(\mathbf{t}_N \mid \mathbf{a}_N) \mathrm{d}\mathbf{a}_N \\ &= \int p(a_{N+1} \mid \mathbf{a}_N) p(\mathbf{a}_N \mid \mathbf{t}_N) \mathrm{d}\mathbf{a}_N \end{aligned}$$

donde se ha usado $p(t_{N+1} \mid a_{N+1}, \mathbf{t}_N) = p(t_{N+1} \mid a_{N+1})$. Si se asume que la función a(x) sigue la forma lineal de la ecuación inicial, se puede demostrar que la función de probabilidad $p(a_{N+1} \mid a_N)$ tiene la forma

$$p(a_{N+1} \mid \mathbf{a}_N) = \mathrm{N}(a_{N+1} \mid \mathbf{k}^{\mathrm{T}} \mathbf{C}_N^{-1} \mathbf{a}_N, c - \mathbf{k}^{\mathrm{T}} \mathbf{C}_N^{-1} \mathbf{k})$$

La integral se puede evaluar encontrando una aproximación de Laplace para la distribución posterior $p(a_N \mid t_N)$ y usando el resultado estándar para la convolución de dos distribuciones gaussianas

La distribución a-priori $p(a_N)$ está dada por un proceso gaussiano de media cero con matriz de covarianza $C_N$ y la verosimilitud de los datos (asumiendo independencia estadística) está dada por

$$p(\mathbf{t}_N \mid \mathbf{a}_N) = \prod_{n=1}^{N} \sigma(a_n)^{t_n} (1 - \sigma(a_n))^{1-t_n} = \prod_{n=1}^{N} \exp(a_n t_n) \sigma(-a_n)$$

La aproximación de Laplace se obtiene haciendo una expansión en serie de Taylor para el logaritmo del término $p(a_N \mid t_N)$. Este logaritmo está dado como

$$\begin{aligned} \psi(\mathbf{a}_N) &= \ln p(\mathbf{a}_N) + \ln p(\mathbf{t}_N \mid \mathbf{a}_N) \\ &= -\frac{1}{2} \mathbf{a}_N^{\mathrm{T}} \mathbf{C}_{N+1}^{-1} \mathbf{a}_N - \frac{N}{2} \ln(2\pi) - \frac{1}{2} \ln \left| \mathbf{C}_{N+1} \right| \\ &\quad + \mathbf{t}_N^{\mathrm{T}} \mathbf{a}_N - \sum_{n=1}^{N} \ln(1 + \exp(a_n)) + \text{const} \end{aligned}$$

El modo de la distribución posterior se puede encontrar del gradiente de la expresión anterior. El gradiente está dado como

$$\nabla \psi(\mathbf{a}_N) = \mathbf{t}_N - \boldsymbol{\sigma}_N - \mathbf{C}_N^{-1}\mathbf{a}_N ,$$

donde $\sigma_N$ es un vector de elementos $\sigma(a_n)$. El valor de $a_N$, que hace cero el gradiente, se encuentra utilizando un proceso recursivo conocido como el algoritmo de mínimos cuadrados recursivos reponderados (IRLS). La fórmula de actualización iterativa para $a_N$ está dada por

$$\mathbf{a}_N^{\text{nuevo}} = \mathbf{C}_N(\mathbf{I} + \mathbf{W}_N\mathbf{C}_N)^{-1}\{\mathbf{t}_N - \boldsymbol{\sigma}_N + \mathbf{W}_N\mathbf{a}_N\} ,$$

donde $W_N$ es una matriz diagonal con elementos $\sigma(a_n)(1-\sigma(a_n))$. En el modo, el gradiente

$$\nabla \psi(\mathbf{a}_N)$$

se desvanece y la solución $a_N^*$ satisface,

$$\mathbf{a}_N^* = \mathbf{C}_N(\mathbf{t}_N - \boldsymbol{\sigma}_N) ,$$

Una vez se encuentra el modo $a_N^*$ del posterior, la matriz hessiana se determina como

$$\mathbf{H} = -\nabla\nabla\psi(\mathbf{a}_N) = \mathbf{W}_N + \mathbf{C}_N^{-1} ,$$

donde los elementos de $W_N$ se evalúan usando $a_N^*$. Lo anterior define una distribución gaussiana que sirve como aproximación a la distribución posterior $p(a_N \mid t_N)$, como

$$q(\mathbf{a}_N) = \mathrm{N}(\mathbf{a}_N|\mathbf{a}_N^*,\mathbf{H}^{-1})$$

Combinando la expresión anterior y evaluando a continuación la integral, se tiene que

$$p(a_{N+1} \mid \mathbf{t}_N) = \mathrm{N}(a_{N+1} \mid \mathbf{k}^{\mathrm{T}}(\mathbf{t}_N - \boldsymbol{\sigma}_N), c - \mathbf{k}^{\mathrm{T}}(\mathbf{W}_N^{-1} + \mathbf{C}_N)^{-1}\mathbf{k})$$

Debido a que $p(a_{N+1} \mid t_N)$ sigue una distribución gaussiana, para evaluar la expresión se puede emplear la aproximación

$$\int \sigma(a)\mathrm{N}(a|\mu,\sigma^2)da = \sigma(\kappa(\sigma^2)\mu)$$

donde $\kappa(\sigma^2) = (1+\pi\sigma^2/8)^{-1/2}$.

Si bien la formulación de procesos gaussianos es consistente desde el punto de vista teórico, la generalización para el caso de clasificadores con múltiples clases no es inmediata, como en el caso de redes neuronales, si se quisieran considerar múltiples tipos de sismos.

Sería interesante explorar formas alternativas de caracterización que permitieran explotar de mejor manera la información contenida en las señales o, de otro lado, considerar la posibilidad de considerar registros tridimensionales

### 2.9 Fundamento teórico del sistema de Transformación de Atributos + Clasificador usado

#### 2.9.1 K-PLS (Kernel + Partial Least Square)

Partiendo de el sistema de k-NN donde se fijaba el número de elementos a considerar. En los métodos de Kernel se fija una distancia h y "votan" todos los elementos que no disten más de ésta. Una vez definida la métrica con una distancia d, el individuo X pertenecerá a la población $P_0$ si

$$\sum_{i=1}^{n} I_{\{Y_i=0, d(\chi_i,\chi)\leq h\}} > \sum_{i=1}^{n} I_{\{Y_i=1, d(\chi_i,\chi)\leq h\}}$$

De forma más general, si definimos una función de Kernel

decreciente, por ejemplo,

$$K : [0,\infty) \longrightarrow [0,\infty)$$

el núcleo gausiano, o

$$K(x) = e^{-x^2}$$

el núcleo de Epanechnikov,

$$K(x) = (1-x^2)I_{[0,1]}$$

$g_n(x)$ asigna X a la clase 0 si

$$\sum_{i=1}^{n} I_{\{Y_i=0\}} K\left(\frac{d(\chi_i,\chi)}{h}\right) > \sum_{i=1}^{n} I_{\{Y_i=1\}} K\left(\frac{d(\chi_i,\chi)}{h}\right)$$

Conforme a la definición aplicada en el apartado 2.8, los métodos de Kernel pueden verse como un método de plug-in definiendo:

$$\eta_n(x) = \frac{\sum_{i=1}^{n} Y_i K\left(\frac{d(\chi_i,\chi)}{h}\right)}{\sum_{i=1}^{n} K\left(\frac{d(\chi_i,\chi)}{h}\right)}$$

Las dificultades de estos métodos radican en la elección de la distancia de un modo similar a **k-NN** (a veces es mejor utilizar semi-métricas $|d(x,x)-0|$ basadas en **PCA** o **PLS** para funciones poco suaves y en la selección de h)

**PLS o Partial Least Square.** Los mínimos cuadrados parciales (PLS) son una técnica de reducción de dimensiones que ha ganado mucha presencia recientemente. La idea que subyace es similar a la de PCA, proyecciones en las direcciones de máxima variabilidad, con la diferencia de que PLS tiene en cuenta la respuesta Y, lo que lo hace mejor que este para clasificación

En el caso funcional, este método cada vez aparece en más referencias con datos funcionales, y aún teniendo su origen en la regresión se aplica cada vez más en clasificación.

PLS es un método originario de la química, en concreto de la quimiometría, parte de la química que aplica métodos matemáticos y estadísticos sobre datos químicos. Se pueden encontrar abundantes referencias en publicaciones como Journal of Chemometrics

Fue planteado por primera vez por Wold en 1975 como método de regresión mediante el algoritmo iterativo NIPALS. Los orígenes del método son empíricos y surgió, como muchos otros, sin un fuerte fundamento teórico detrás.

Sin embargo, los buenos resultados obtenidos hasta el momento han motivado su estudio formal

El objetivo principal de dicho método es maximizar la covarianza con la clase definida de la forma

estándar:

$$Cov^2(X,Y) = E[(X - E(X))(Y - E(Y))]^2 = (\frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})(y_i - \overline{y}))^2$$

Según este planteamiento se trata al vector Y, por ejemplo de ceros y unos (en el caso general podrá ser una matriz) como una variable de regesión y se implemente el algoritmo NIPALS. De este modo se buscarán las K componentes que minimicen el error de:

$$\widehat{Y} = \beta_0 + \beta_1 T_1 + .... + \beta_K T_K$$

Ahora el objetivo es la clasificación, por lo que el algoritmo no devolverá unos modelos de regresión, sino un conjunto de componentes (proyecciones), T y una matriz de proyección $Z$ para tratar los nuevos datos

El algoritmo para el caso de salidas unidimensionales fue concluido por Trygg modificando la idea de Wold:

1. Inicialización: $Y(0) = Y$ estandarizado y $X(0) = X$ estandarizado.
2. Para k de 1 a d:
3. $w(k) = Cov[Y(k-1), X(k-1)]$
4. $w(k) = \frac{w(k)}{\|w(k)\|}$
5. $T_k = X(k-1)w(k)$
6. $v(k) = (\frac{T_k^t Y(k-1)}{T_k^t T_k})$ ; $b(k) = (\frac{T_k^t X(k-1)}{T_k^t T_k})$
7. $Y(k) = Y(k-1) - T_k v(k)$ ; $X(k) = X(k-1) - T_k b(k)$
8. Si $k = 1$ $z_1 = w(1)$ y si no $z_k = [Id - \sum_{j=1}^{k-1} z_j b(j)]w(k)$

Los indices entre paréntesis hacen referencia a la iteración mientras que los subindices señalan las coordenadas. Este algoritmo extrae en cada rearación una nueva componente ortogonal a las anteriores de forma que se maximiza la función objetivo eligiendo como vector de proyección la covarianza entre los datos de entrada y la clase. Al final de cada iteración se actualizan las matrices X y Y eliminando la "información" obtenida con la nueva componente generada, esto se, se calculan los coeficientes de regresión asociados a cada una de ellas y se restan.

Según este planteamiento, las componentes $T_i$ son las proyecciones de los datos. La matriz $Z$ es la matriz de proyección para nuevos datos.

PLS adaptado a un sistema de reducción de dimensiones orientado a clasificación irrumpe con fuenza en el ámbito de la biomformática para tratar los problemas de microarrays

La superioridad de PLS sobre PCA viene determinada por que PLS intenta maximizar la covarianza de las proyecciones de los datos con la clase a la que pertenecen. Aquí es donde se encuentra la diferencia fundamental con PCA ya que PLS tiene en cuenta la información que da la clase, mientras que PCA contempla únicamente la varianza global de los datos

En general, las salidas tendrán dimensión M, si bien, para hacer el estudio de clasificación se fija M=1 (problema binario) para después generalizar. De esta forma,

$$T_{NxD} = (T_1, \cdots, T_D)$$

tiene por columnas las componentes extraídas por PLS, que son ortonormales. De todas formas, considerando la matriz completa T no ganamos nada, ya que se vuelve a un espacio de dimensión D. Por este motivo se cogen sólo las primeras K componentes, K<<<<D.

Como en la mayoria de los modelos de este tipo, la elección de K es uno de los mayores problemas y no existe una solución general. Este parámetro suele ajustarse en la gase de validación por sistemas estándar como la validación cruzada

En el caso de PLS el problema es interesante porque permite ser abordado tanto por un algoritmo iterativo que va extrayendo las componentes una a una en función de las anteriores, como un problema de valores propios

$$(a_{k+1}, b_{k+1}) = \underset{a \in \mathbb{R}^D, b \in \mathbb{R}^M, a^t A = 0}{argmax} \frac{Cov(a^t X, b^t Y)^2}{(a^t a)(b^t b)}$$

Donde ak es la k-ésima componente correspondiente al k-ésimo autovalor, y A es la matriz de las k-1 componentes ya elegidas. Las matrices a diagonalizar serían

$$H = \Sigma_{XY}\Sigma_{YX} = \begin{pmatrix} Cov(X_1,Y)^2 & Cov(X_1,Y)Cov(X_2,Y) & \cdots & Cov(X_1,Y)Cov(X_D,Y) \\ Cov(X_2,Y)Cov(X_1,Y) & Cov(X_2,Y)^2 & \cdots & Cov(X_2,Y)Cov(X_D,Y) \\ & \cdots & & \\ Cov(X_D,Y)Cov(X_1,Y) & Cov(X_D,Y)Cov(X_2,Y) & \cdots & Cov(X_D,Y)^2 \end{pmatrix}$$

en el caso de M=1, y en general se tendrá en la diagonal la suma de las covarianzas al cuadrado de cada atributo con todas las salidas ya que:

$$\Sigma_{XY} = \begin{pmatrix} Cov(X_1,Y_1) & Cov(X_1,Y_2) & \cdots & Cov(X_1,Y_M) \\ Cov(X_2,Y_1) & Cov(X_2,Y_2) & \cdots & Cov(X_2,Y_M) \\ & \cdots & & \\ Cov(X_D,Y_1) & Cov(X_D,Y_2) & \cdots & Cov(X_D,Y_M) \end{pmatrix}$$

y

$$\Sigma_{YX} = \Sigma_{XY}^t$$

su traspuesta. En este caso, las componentes se sacarían resolviendo el problema de autovalores hasta la dimensión K que queramos.

Por último, hay que comentar que las limitaciones de PLS en cuanto al número de componentes. Con el enfoque de valores propios, PLS tiene el problema de que si hay C clases solo se pueden obtener C-1 autovalores distintos de cero debido al rango de las matrices

$$\Sigma_{YX} = \Sigma_{XY}^t$$

Si se requieren más proyecciones se puede utilizar la versión iterativa. En este caso el límite será el rango de X (igual que en PCA).

### 2.10 Estimación de la bondad de un clasificador

Considerando que se el problema planteado en el presente trabajo se trata de una regresión, se considera que uno de los métodos para estimar la bondad de un clasificador será el cálculo del error cuadrático medio.

El criterio razonable para escoger un determinado estimador de un parámetro $\theta$ es tomar aquel que cometa, en promedio, el menor error en la estimación. Como, en principio, se pretende penalizar

igualmente los errores por defecto que por exceso se podría establecer como cantidad a minimizar la esperanza de la diferencia entre el estadístico $T$ y el parámetro $\theta$ (en valor absoluto para impedir que los errores por defecto y por exceso se anulen mutuamente)

$$E\left[\left|T \quad \theta\right|\right]$$

Aunque este operador resulta razonable, presenta el inconveniente de que la función valor absoluto es complicada de manejar desde un punto de vista matemático. Por dicha razón suele utilizarse el *error cuadrático medio* (ECM) de un estimador $T$, definido como sigue:

$$E\left[(T-\theta)^2\right]$$

Una propiedad interesante del ECM es que puede descomponerse como la suma de dos componentes: la varianza del estimador más su sesgo al cuadrado

[redacted]

Por tanto, en el caso de comparar diversos estimadores centrados de un parámetro $\theta$, el ECM coincidirá con sus varianzas. Con lo que el estimador con menor ECM coincidirá con el de menor varianza

Debe quedar claro, sin embargo, que no se debe emplear ECM únicamente para evaluar la bondad de los sistemas de clasificación, sino que tanto más importante que el ECM, es apoyar los resultados con otros cálculos y representaciones gráficas donde se puedan extraer conclusiones sobre la distribución de los errores.

Una gráfica especialmente útil cuando los atributos son numéricos es la gráfica de dispersión (Scatterplot). Esta gráfica no tiene el sesgo de los intervalos numéricos.

En estas gráficas se muestra una tercera dimensión. En el presente proyecto, se presentan en un eje X la temperatura efectiva real, en el eje Y la predicción de temperatura efectiva y, marcando el degradado por colores etiquetamos la diferencia en la aproximación.

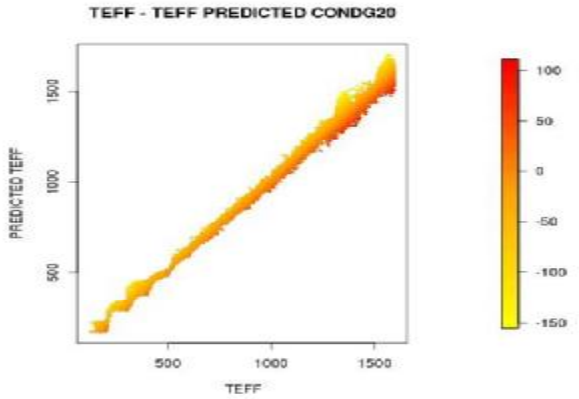


Figura 12 : Ejemplo de Gráfica de dispersion: Teff vs Teff predicted, escala de color nos determina la desviación

Otro tipo de gráficas de dispersión empleadas en el presente proyecto representan gráficas teff-logg donde, de la misma forma que los anteriores, el código de color que refleja el error en la estimación de temperatura efectiva

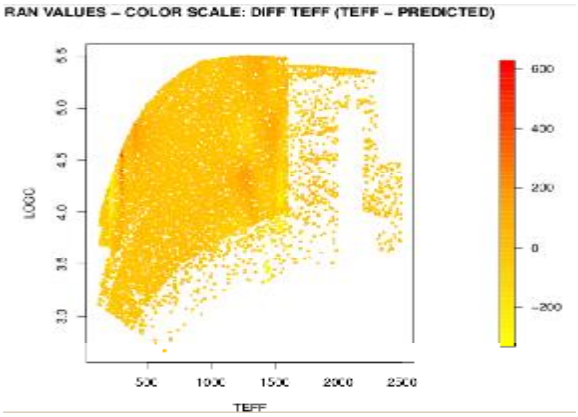


Figura 12 : Ejemplo de Gráfica de dispersión: Teff vs Log , el gradiente de color muestra la diferencia entre teff y teff predicted

Téngase en cuenta que el degradado de color muestra el error en la predicción de la temperatura con respecto a la real.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real

## 2.11. Herramientas empleadas

### 2.11.1 Sistema Operativo Ubuntu

Es el sistema operativo sobre el que se ha desarrollado el trabajo. Se ha seleccionado debido a las capacidades de manipulación de los espectros fue Linux. En concreto Ubuntu

Ubuntu es una distribución GNU/Linux basada en Debian GNU/Linux que proporciona un sistema operativo actualizado y estable para el usuario medio, con un fuerte enfoque en la facilidad de uso e instalación del sistema

Al igual que otras distribuciones se compone de múltiples paquetes de software normalmente distribuidos bajo una licencia libre o de código abierto. Estadísticas web sugieren que el porcentaje de mercado de Ubuntu dentro de las distribuciones Linux es de aproximadamente 50%, y con una tendencia a subir como servidor web.

Está patrocinado por Canonical Ltd., una compañía británica propiedad del empresario sudafricano Mark Shuttleworth que en vez de vender la distribución con fines lucrativos, se financia por medio de servicios vinculados al sistema operativo y vendiendo soporte técnico

Además, al mantenerlo libre y gratuito, la empresa es capaz de aprovechar los desarrolladores de la comunidad en mejorar los componentes de su sistema operativo. Canonical también apoya y proporciona soporte para cuatro derivaciones de Ubuntu: Kubuntu, Xubuntu, Edubuntu y la versión de Ubuntu orientada a servidores (*Ubuntu Server Edition*)

La pagina oficial es http://www.ubuntu.com

### 2.11.2 Paquete Software de Minería de Datos Weka

Para abordar este cometido se apoyara en Weka, un conjunto de librerías de algoritmos de aprendizaje para tareas de Data Mining escritas en JAVA. Es un software que ha sido desarrollado en la universidad de Waikato (Nueva Zelanda) bajo licencia GPL.

El paquete Weka contiene una colección de herramientas de visualización y algoritmos para análisis de datos y modelado predictivo, unidos a una interfaz gráfica de usuario para acceder fácilmente a sus funcionalidades

La versión original de Weka fue un *front-end* en TCL/TK para modelar algoritmos implementados en otros lenguajes de programación, más unas utilidades para preprocesamiento de datos desarrolladas en C para hacer experimentos de aprendizaje automático

Esta versión original se diseñó inicialmente como herramienta para analizar datos procedentes del dominio de la agricultura, pero la versión más reciente basada en Java (WEKA 3), que empezó a desarrollarse en 1997, se utiliza en muchas y muy diferentes áreas, en particular con finalidades docentes y de investigación

Se ha empleado Weka para la predicción empleando clasificadores como máquinas de vectores soporte, procesos gausianos, K vecinos cercanos. También se ha empleado para aplicar la transformación PCA más los algoritmos comentados anteriormente a través del clasificador "AttributeSelectedClassifier".

### 2.11.3 Paquete Software estadístico R

El paquete software estadístico R (también conocido como GNU S) desarrollado por Bell Laboratories by John Chambers, es empleado en determinadas tareas del presente proyecto para la manipulación de los diferentes conjuntos de datos : aplicación de normalización en area1, estudio de sistemas de reducción de ruido como wavelets o moving average, aplicación del filtro kernel, estudio de la transformación Partial Least Square, obtención de gráficas comparativas de la bondad de los modelos de clasificación obtenidos (scatterplots, plotters, . . )

Las ventajas de uso de R incluyen:

- La capacidad de combinar, sin fisuras, análisis "preempaquetados" (ej., una regresión logística) con análisis ad-hoc, específicos para una situación; capacidad de manipular y

modificar datos y funciones.

- Los gráficos de alta calidad (revelaciones de la visualización de datos y producción de gráficas para papers)

R se distribuye con licencia GNU GPL o General Public License (ver http://www.gnu.org/licenses/gpl.html .)

La GPL no pone ninguna restricción al uso de R. Restringe su distribución (ha de ser GPL)

R se obtiene en http://cran.r-project.org

R consta de un "sistema base" y de paquetes adicionales. en el presente proyecto hemos usado los siguientes: "wmtsa", "wavelets", "pls", "princomp", "fields".

## 3. El Proyecto: Fase de Experimentación

### 3.1. Experimentos de preprocesado

Por lo justificado y comentado en el apartado 2.4 sobre los orígenes de datos, de los conjuntos de datos tanto para entrenamiento como para validación facilitados por DPAC CU2/CU8, emplearemos los espectros del fotómetro rojo (RP) compuesto de 180 atributos (flujos medidos en diferentes longitudes de onda) que definen el espectro de cada candidata a Enana Ultrafría

Descartaremos los espectros del fotómetro azul (BP), ya que considera que no serán adecuados a la tarea de regresión debido a la escasa luminosidad de las enanas ultrafrías en las longitudes de onda corta (tal y como se hace visible en la Figura 5).

Se realiza una visualización del espectro electromagnético de los datos nominales, la muestra escogida para la representación ha sido elegida de forma aleatoria en los conjuntos de espectros COND y DUST para G8 y G15. Esta representación se visualiza en las Figuras 14, 15, 16 y 17.

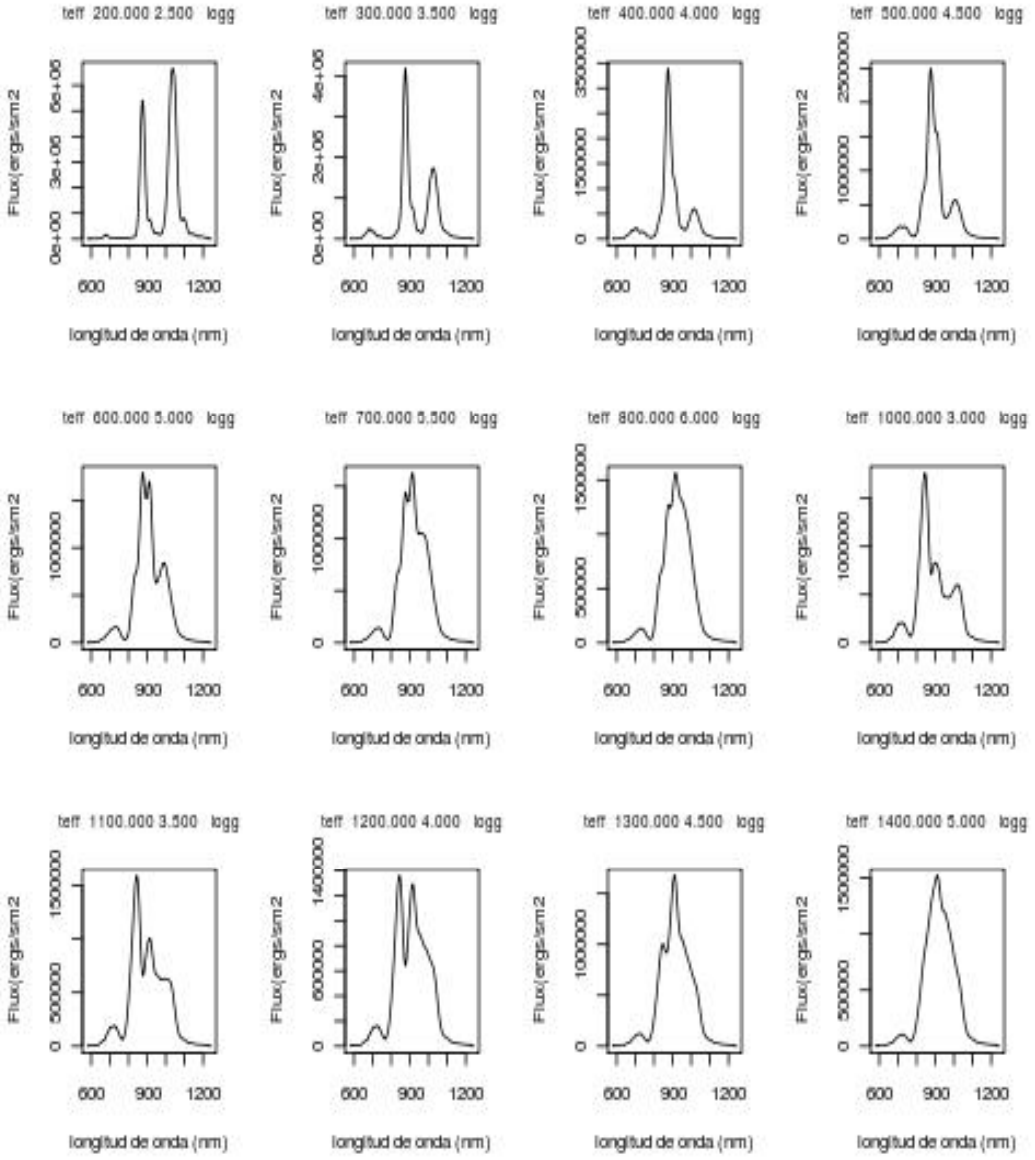


Figura 14: 16 espectros electromagnéticos escogidos aleatoriamente sobre el conjunto de datos

NOM COND para magnitud G8

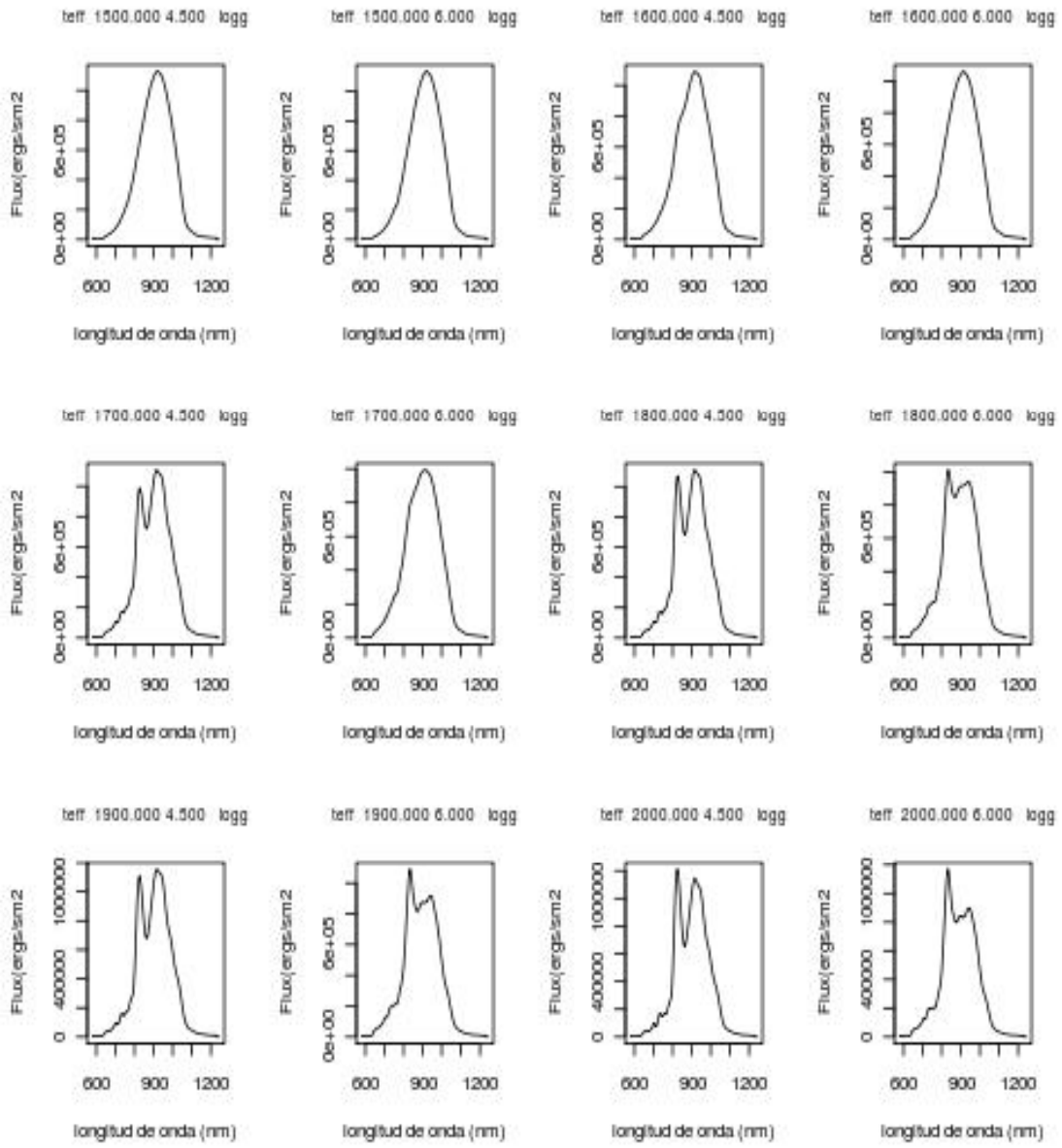


Figura 15: 16 espectros electromagnéticos escogidos aleatoriamente sobre el conjunto de datos

NOM DUST para magnitud G8

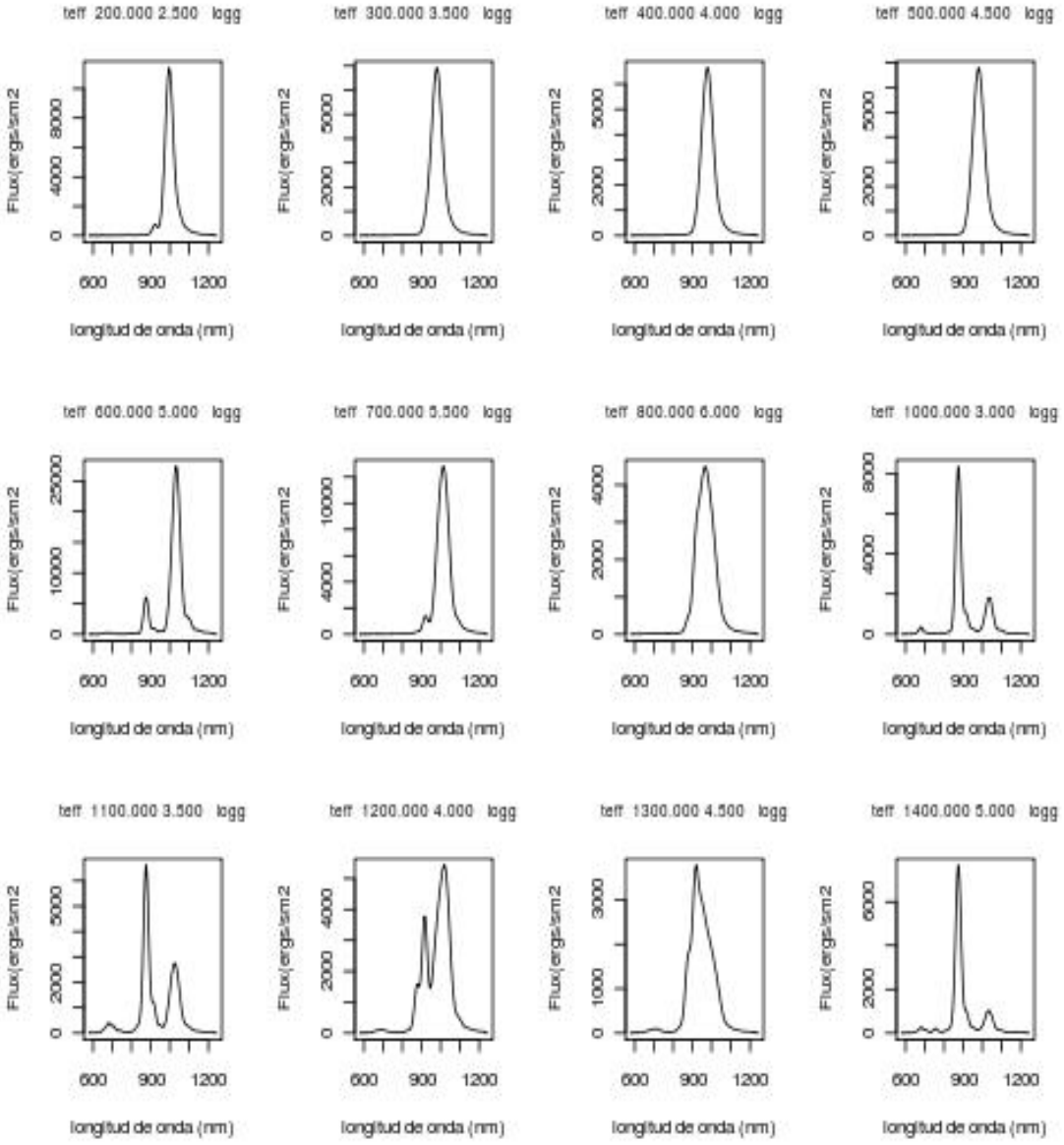


Figura 16: 16 espectros electromagnéticos escogidos aleatoriamente sobre el conjunto de datos

NOM COND para magnitud G15

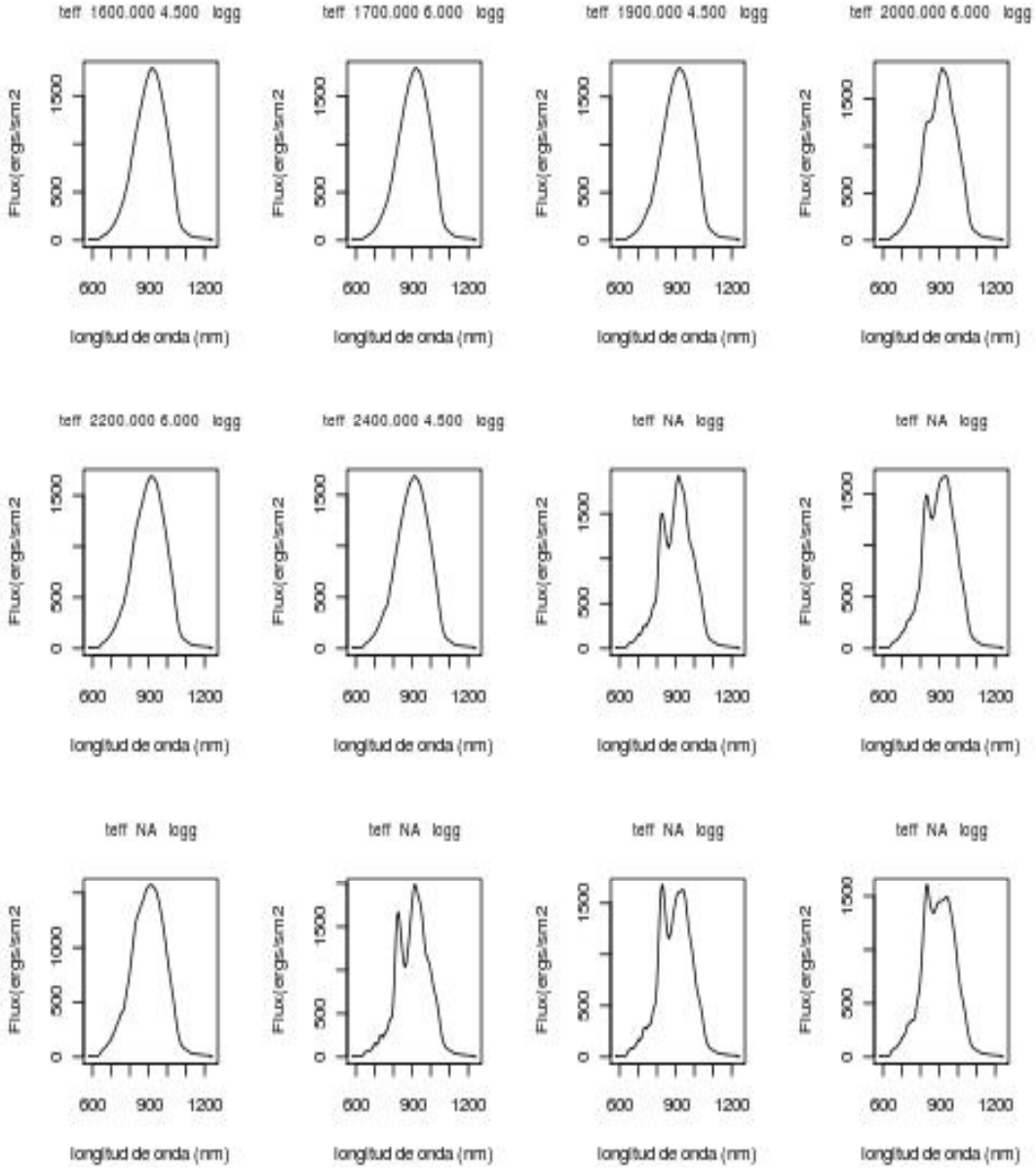


Figura 17: 16 espectros electromagnéticos escogidos aleatoriamente sobre el conjunto de datos

NOMDUST para magnitud G15

En las cuatro figuras anteriores puede extraerse una conclusión común, las variables de cada espectro se encuentran muy correlacionadas, lo que nos hace presuponer que una reducción de dimensionalidad, puede beneficiarnos en la obtención de mejores resultados con los clasificadores

### 3.1.1 Estudio de Normalización

El trabajo fin de máster se ha orientado a la determinación del mejor sistema de normalización, sobre los espectros RP, por lo ya anteriormente comentado.

Simplemente observando los espectros escogidos al azar para las diferentes magnitudes aparentes (G8 y G15), mostrados en las figuras 14, 15, 16 y 17 observamos la diferencia de escala de los valores representados incluso en espectros de la misma magnitud.

La normalización nos debe trasportar los datos a una misma escala. La normalización más común es la normalización lineal uniforme normalizando entre cero y uno teniendo en cuenta la siguiente formula:

$$v' = \frac{v - min}{max - min}$$

Sin embargo, se puede observar como este tipo de normalización no es tolerante a ruido ya que es muy dependiente de los valores mínimos y máximos.

Se han propuesto dos modelos de normalización para los conjuntos de datos. la euclidea implementada directamente en Weka (código comentado en el Anexo *7.3.1 Obtención de Normalización Euclidea*) y Areal implementada en R (código desarrollado en el Anexo *7.2.1 Representación gráfica de los espectros electromagnéticos*)

La representación de los espectros electromagnéticos de los conjuntos de datos una vez aplicada la normalización se puede observar en las figuras 18, 19 y 20

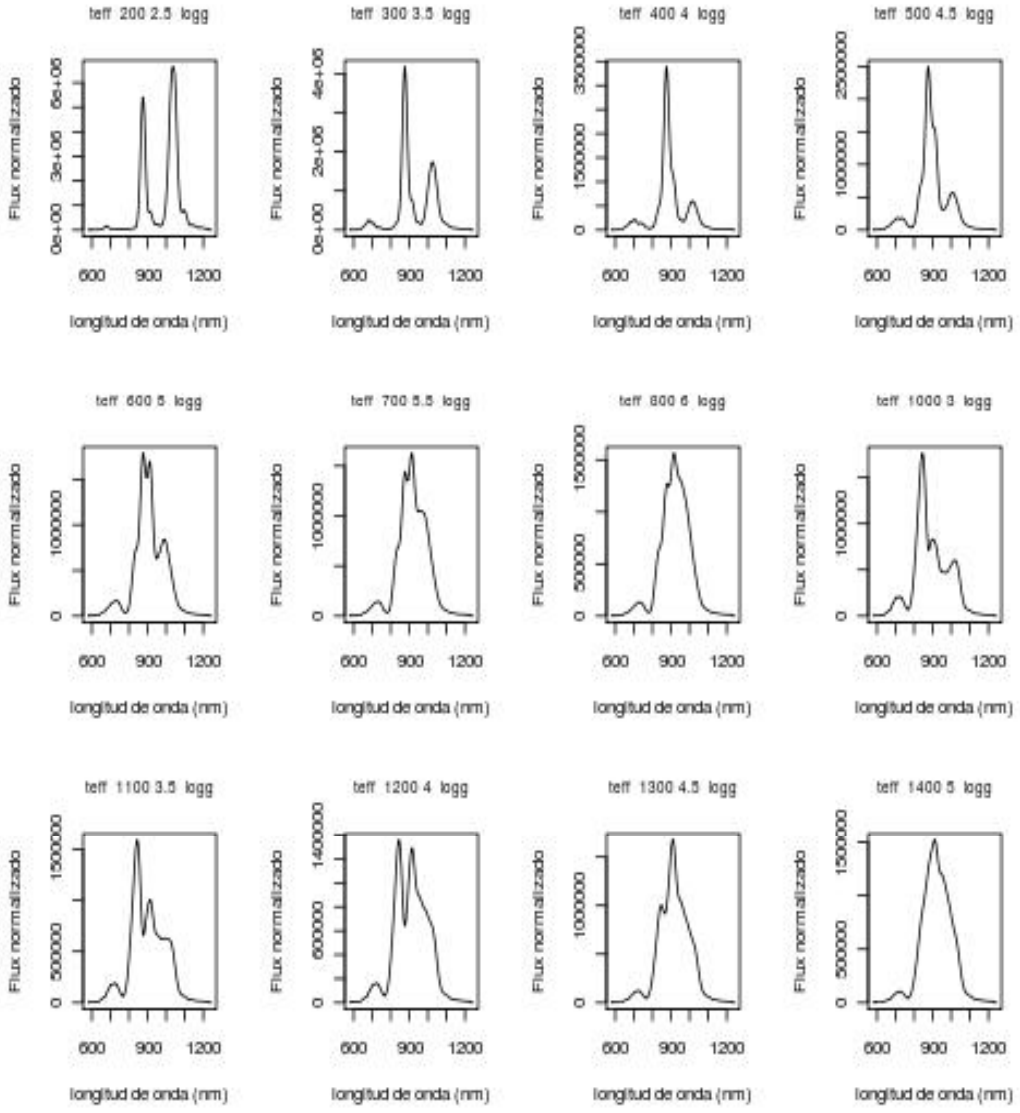


Figura 18: 16 espectros electromagnéticos escogidos aleatoriamente sobre el conjunto de datos NOM. Visualmente se muestran diferentes escalas para distintos espectros. Los datos no están en la misma magnitud.

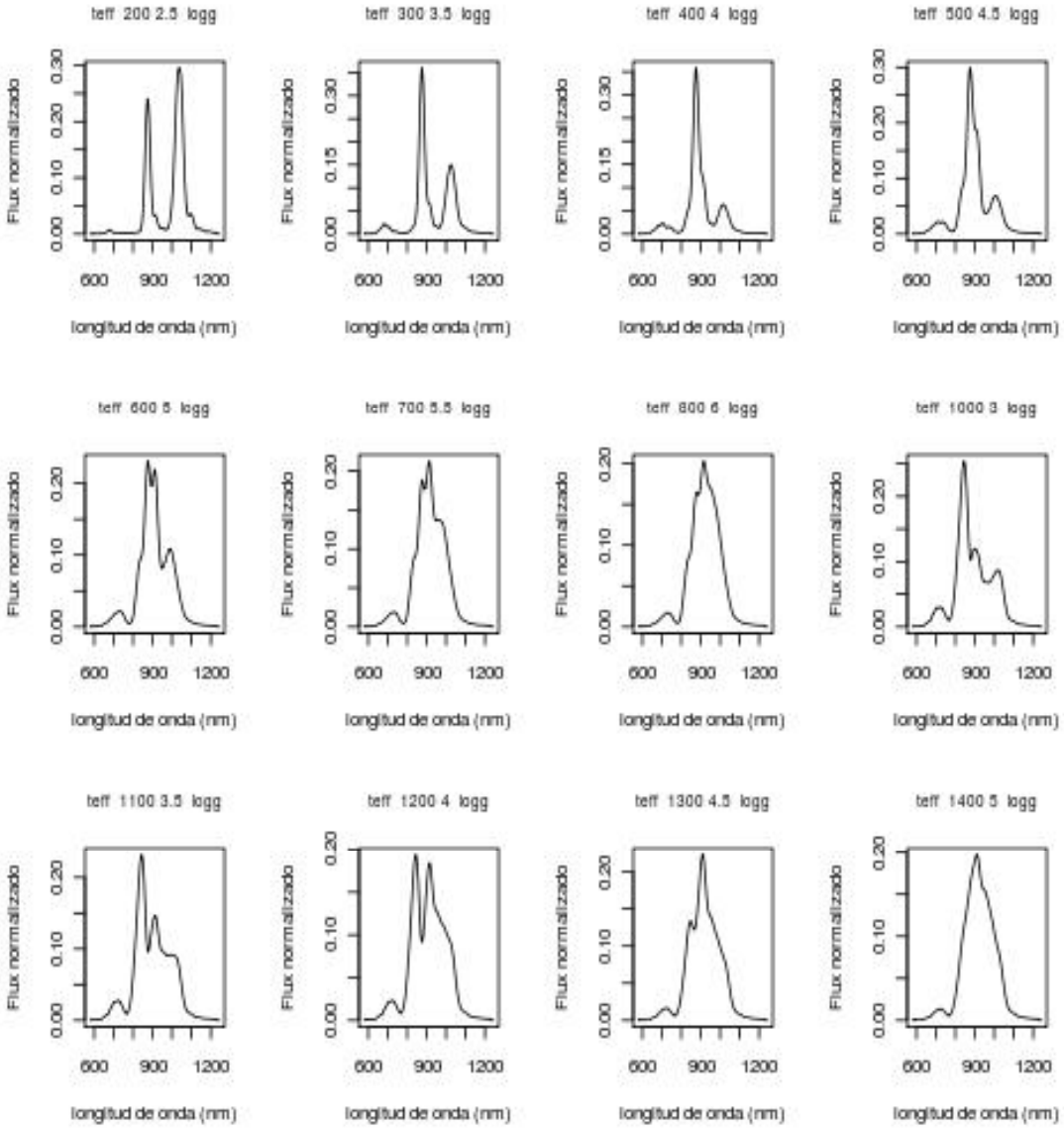


Figura 19: Los mismos 16 espectros electromagnéticos de la figura 18, aplicada la normalización euclídea. Visualmente se observa como los datos se encuentran normalizados al mismo rango.

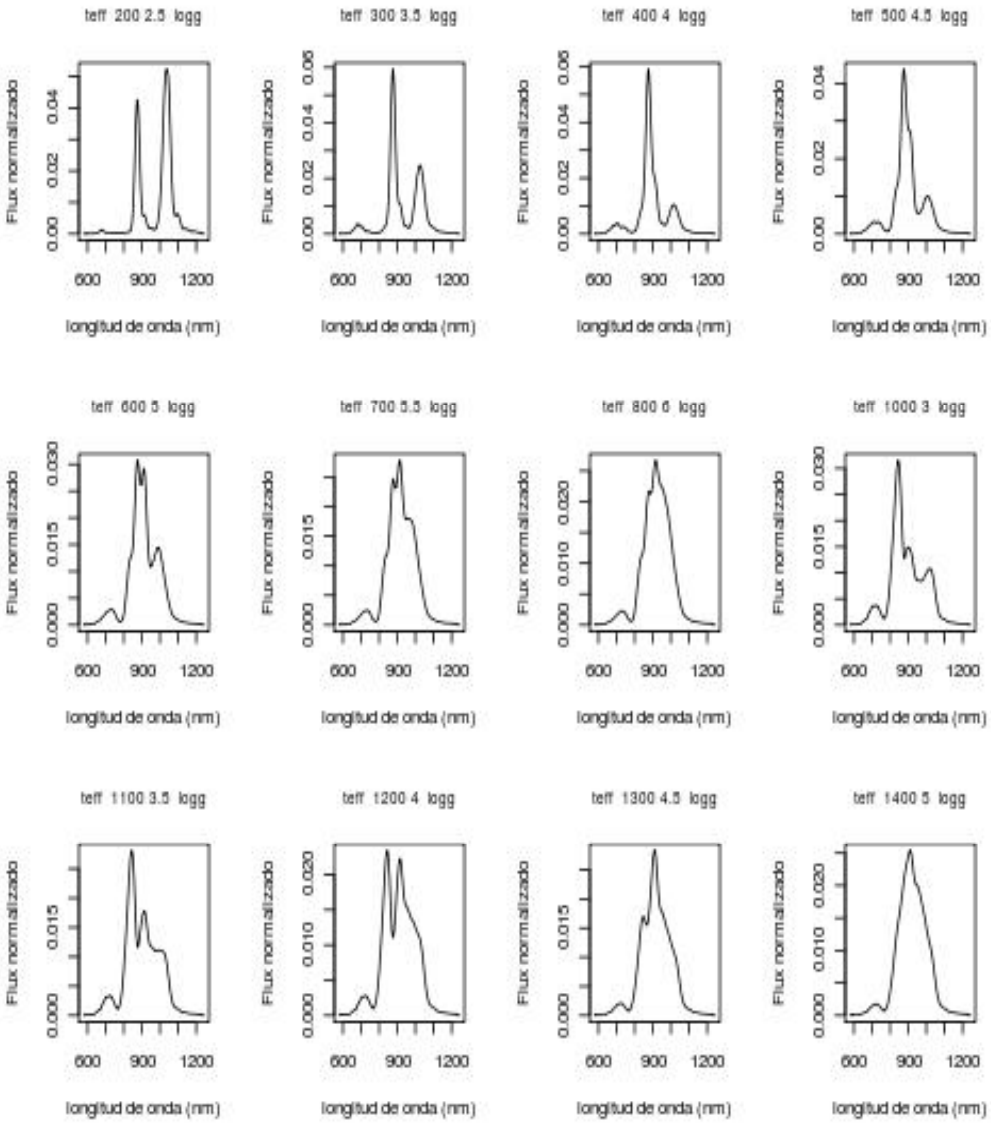


Figura 20: Los mismos 16 espectros electromagnéticos de la figura 18, aplicada la normalización a área1. Visualmente se observa como los datos se encuentran normalizados al mismo rango.

Se comprueba como para un mismo conjunto de datos NOM, se ha conseguido trasladar el espectro a una escala controlada para todos los espectros, manteniendo información

Se realizaron diferentes experimentos iniciales entrenando con los conjuntos de entrenamiento NOMeuclidea y NOMareal, y validando con los conjuntos de datos G18euclidea, G18areal, G19euclidea, G19areal y G20euclidea, G20areal (obtenidos partiendo de los conjuntos de datos presentados en la tabla 1) concluyendo que a la normalización a areal afectaba en menor medida el ruido

### 3.1.2 Estudio de Suavizado del Ruido

Otros experimentos iniciales, generados a partir de la determinación de la normalización a areal consistieron en determinar el mejor método de suavizado del ruido que el sistema se encontraría con el conjunto de espectros recibidos a la salida de la sonda

Se decidió discriminar en la medida de lo posible los conjuntos de espectros para validación, por ese motivo, a partir de este momento, sobre el conjunto de validación se dispone de dos grupos diferenciados: COND y DUST.

Otro detalle que se observa en esta fase de la experimentación previa es que se dispone de un número de espectros elevado. En un principio, la idea fue generar 10 replicas por cada espectro existente de RAN (para G8, G11 y G15), lo cual considerando el tamaño de RAN nos supondría manejar 330 000 espectros en validación. De los cuales 300 000 son para el conjunto de validación COND y 10.000 para DUST (tal y como se observa en la tabla 1)

Cuando se realizaron las primeras pruebas de experimentación, se decidió recortar el conjunto de datos de validación debido al retardo en la obtención de resultados de cada experimento.

Finalmente, se opta por generar una única réplica de ruido a diferente magnitud partiendo de los espectros de RAN para la magnitud aparente G15

De esta forma el conjunto de datos de validación se reduce considerablemente, ya no solo por el espacio que ocupan en disco sino también por el tiempo de realización de la experimentación.

Por lo tanto, los conjuntos de ruido para validación quedan compuestos tal y como se describen en la tabla 2, quedando determinado el tamaño de los conjuntos de datos COND a 10,000 espectros y DUST en 1,000.

También hay que reducir el número de conjuntos de validación, se decide en esta fase, eliminar del estudio de experimentación del suavizado del ruido, los conjuntos de entrenamiento con menos ruido, quedando definidos en la tabla 4 los conjuntos de datos más ruidosos, empleados en esta fase de la experimentación:

| Conjunto de datos | COND | DUST | Nº Espectros | Con ruido? |
|---|---|---|---|---|
| CONDRANX2 | X | | 10000 | SI |
| DUSTRANX2 | | X | 1000 | SI |
| CONDRANX5 | X | | 10000 | SI |
| DUSTRANX5 | | X | 1000 | SI |
| CONDRANX10 | X | | 10000 | SI |
| DUSTRANX10 | | X | 1000 | SI |

Tabla 5 conjuntos de datos seleccionados para la experimentación de suavizado

En este momento, es importante disponer de una visualización de varios espectros aleatorios superpuestos sobre los originales de magnitud aparente G15 para apreciar el ruido introducido

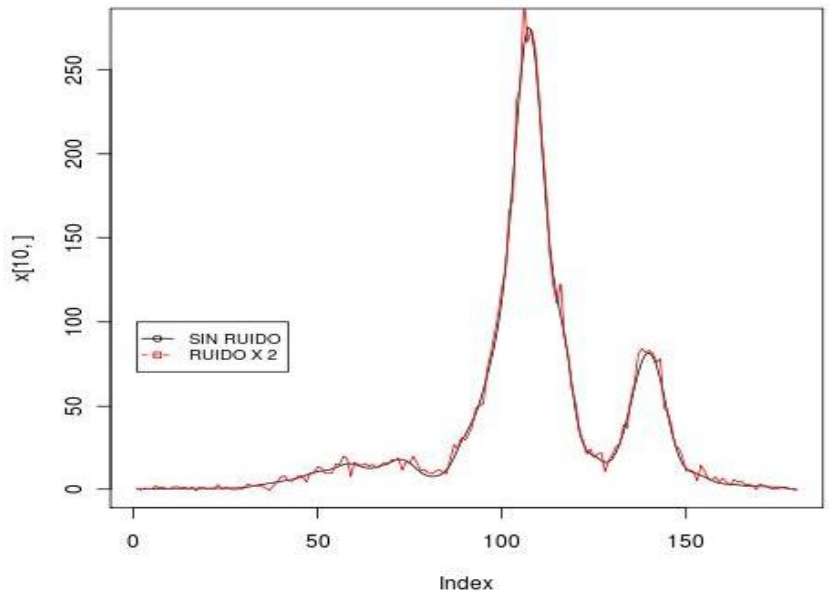


Figura 21 : espectro superpuesto sin ruido COND (en negro) y con ruido generado como RUIDOCONDX2 (en rojo)

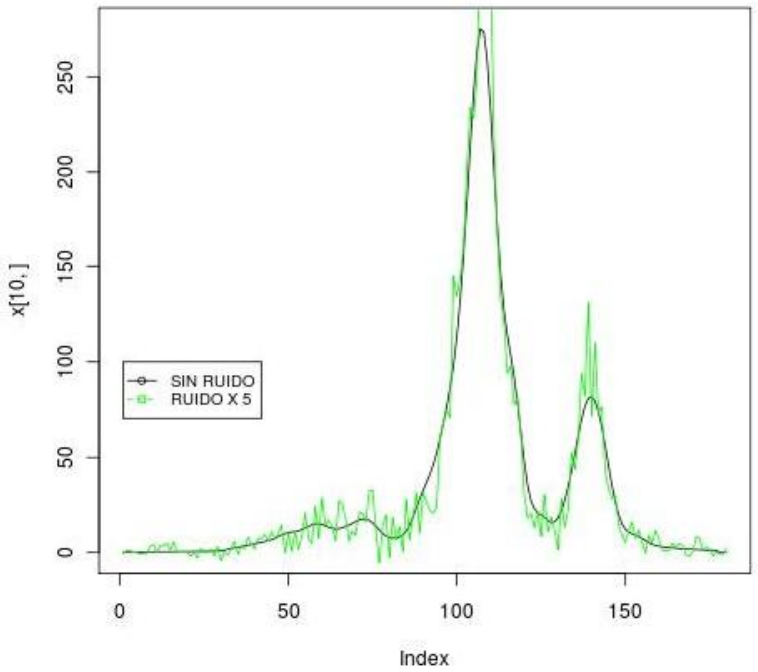


Figura 22 : espectro superpuesto sin ruido COND G15 (en negro) y con ruido generado como RUIDOX5 (en verde) antes de normalizar

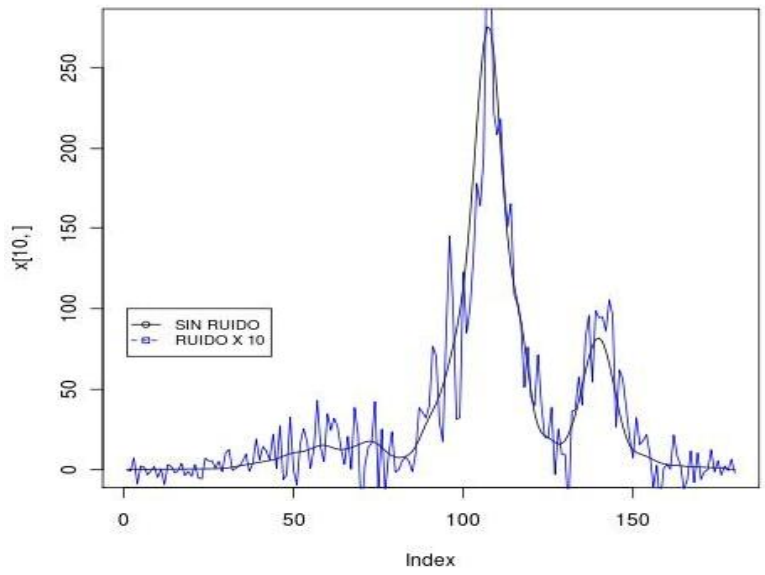


Figura 23 : espectro superpuesto sin ruido COND G15 (en negro) y con ruido generado como RUIDOCONDX10 (en azul) antes de normalizar

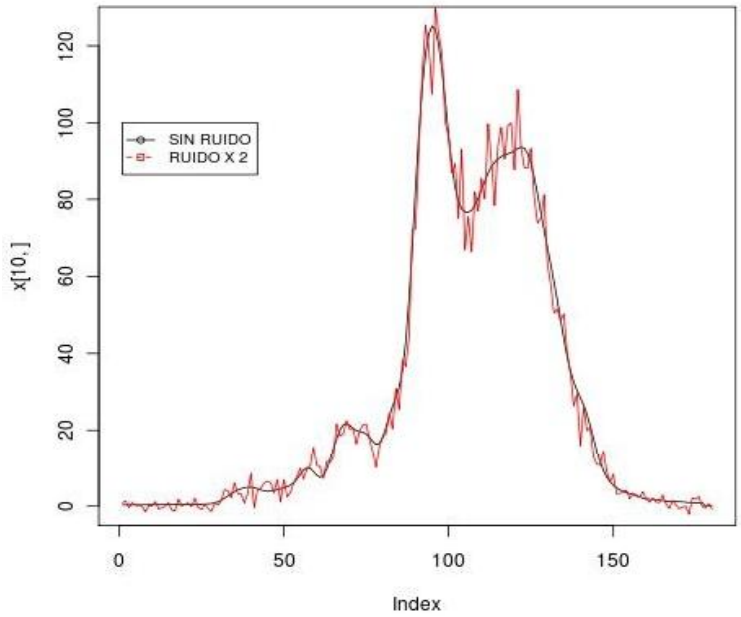


Figura 24 : espectro superpuesto sin ruido DUST G15 (en negro) y con ruido generado como RUIDODUSTX2 (en rojo) antes de normalizar

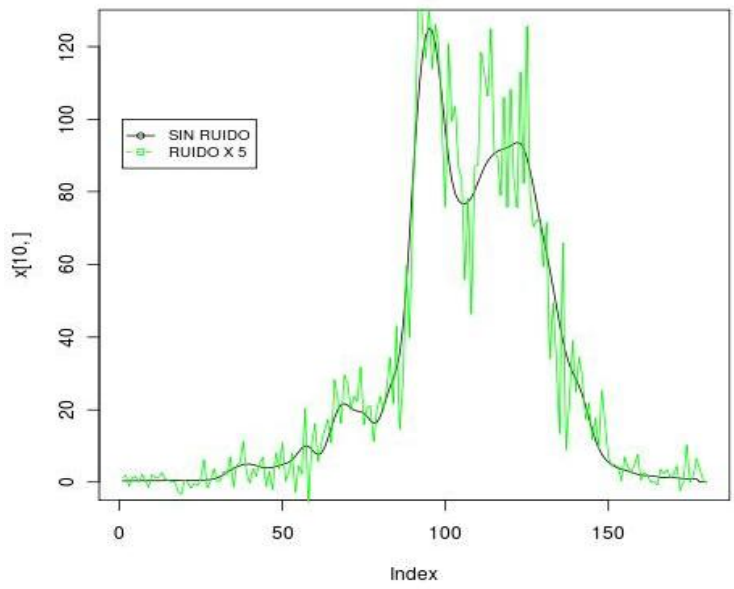


Figura 25 : espectro superpuesto sin ruido COND G15 (en negro) y con ruido generado como RUIDOX5 (en verde) antes de normalizar

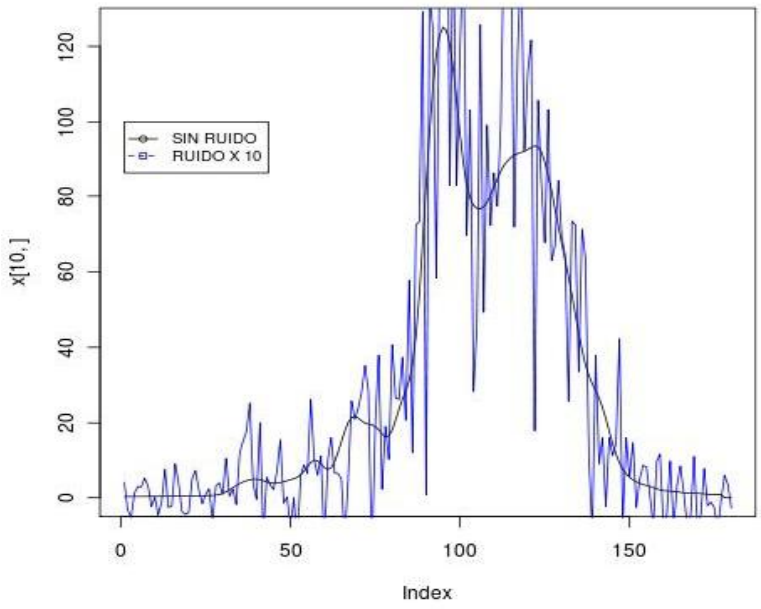


Figura 26 : espectro superpuesto sin ruido COND G15 (en negro) y con ruido generado como RUIDOX10 (en azul) antes de normaliza

Observando las figuras 27 y 28 se demuestra que la aplicación de Wavelets (con diferentes filtros: Daubechies, Best Located, Least Asymetric, Coiflet de diferente orden) y el código de moving average, consigue suavizar el ruido

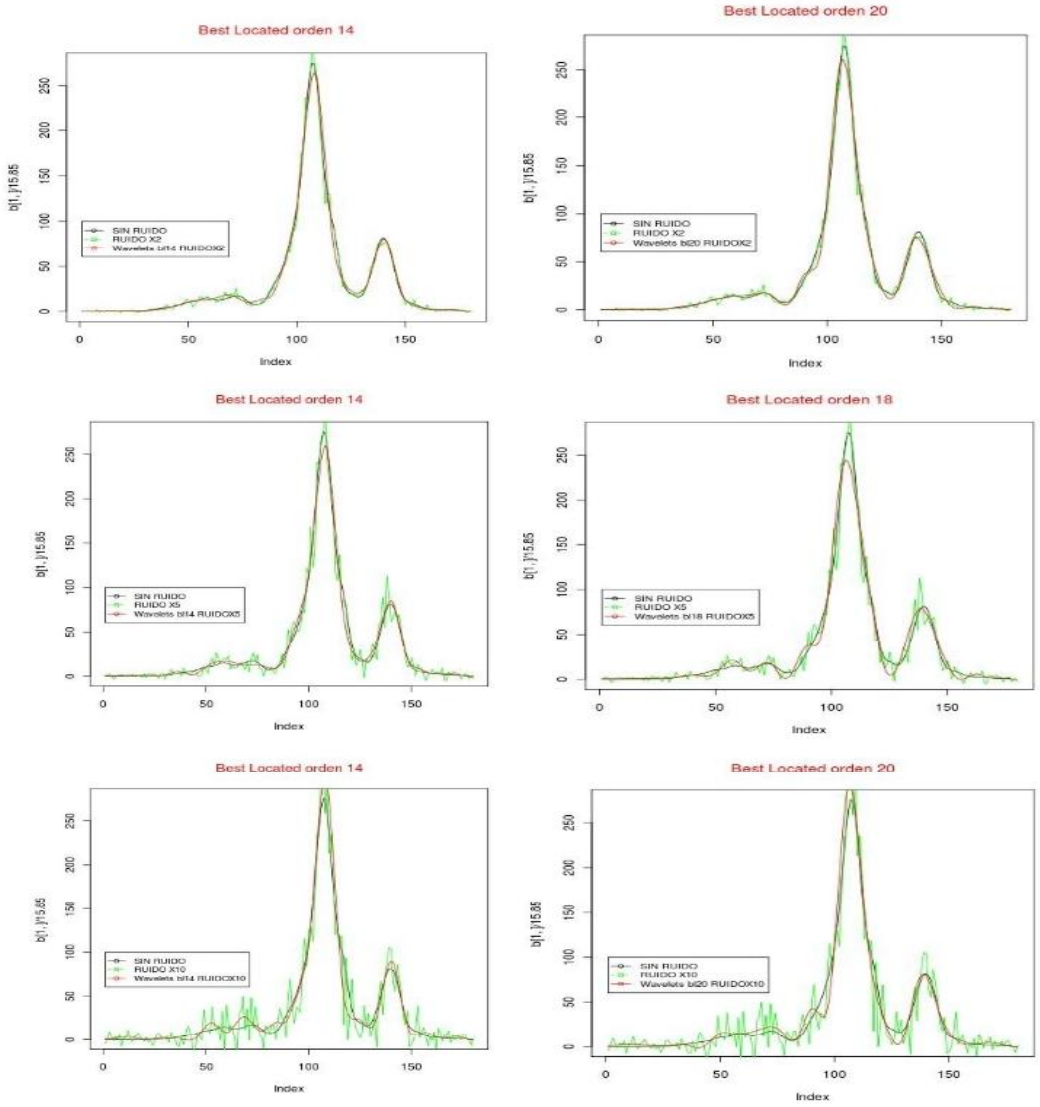


Figura 27: Un ejemplo del suavizado wavelet con filtro Best Located y ordenes 14, 16, 18 y 20 para los conjuntos de datos RUIDOCONDX2, RUIDOCONDX5 y RUIDOCONDX10

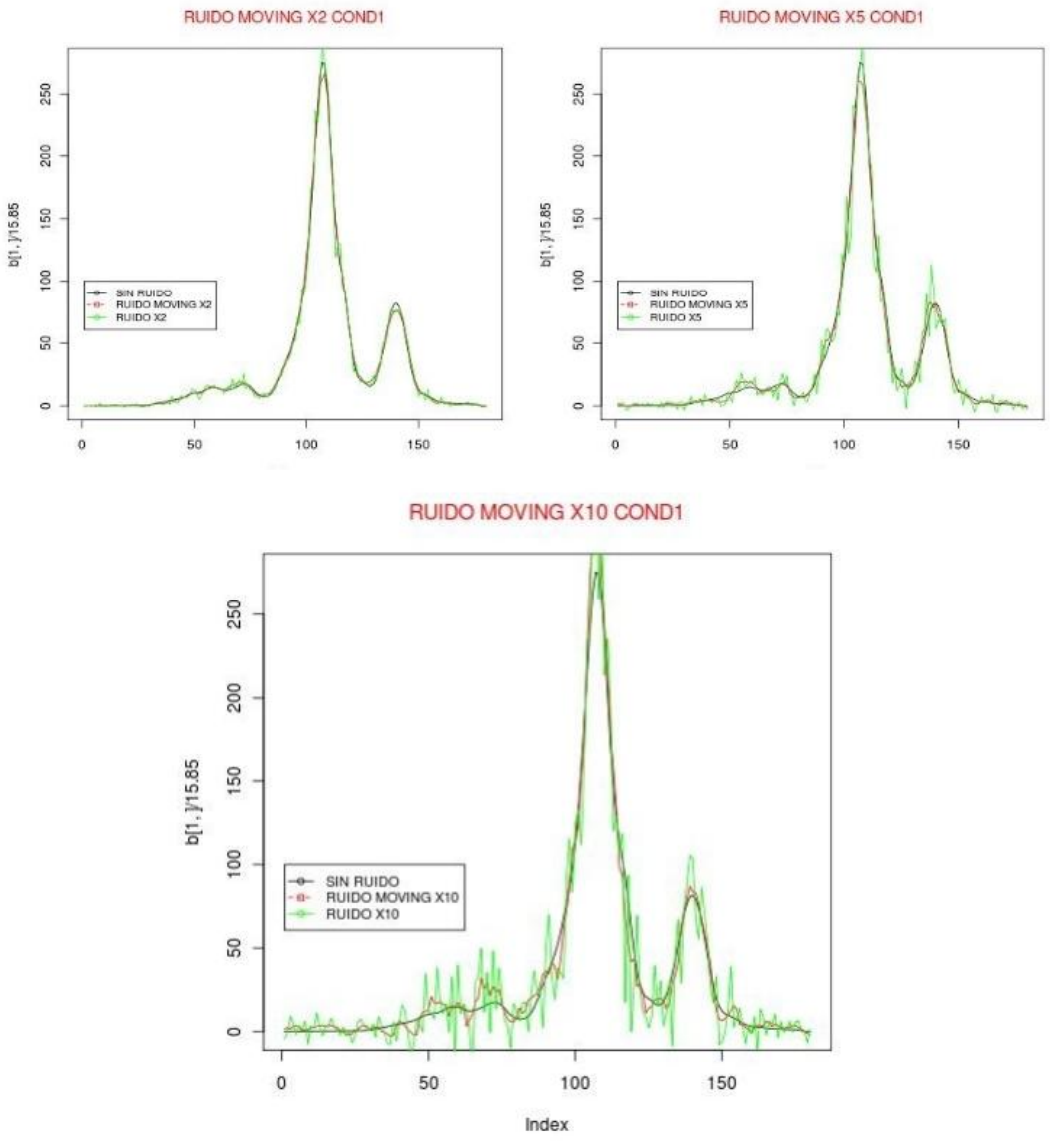


Figura 28: Un ejemplo del suavizado con moving average para los conjuntos de datos RUIDOCONDX2, RUIDOCONDX5 y RUIDOCONDX10

Se aplicaron todas las posibles trasformadas wavelets (filtros y órdenes) sobre todos los conjuntos de espectros comentados

La evaluación de la bondad del suavizado se realizó calculando la media, mediana y sumatorio de los errores cuadráticos medios con respecto a CONDG15 y DUSTG15 para todos los conjuntos de datos RUIDOCONDX2, RUIDOCONDX5, RUIDOCONDX10, RUIDODUSTX2,

RUIDODUSTX5 y RUIDODUSTX10, apoyándose en las representaciones gráficas de los espectros

Una vez calculados para todos los filtros wavelets y en vista de los resultados obtenidos, se decide emplear los sistemas de suavizado mediante wavelets: Coiflet de orden 18 y el filtro Best Located de orden 14.

En todos los casos, (COND y DUST para RUIDOX2, RUIDOX5 y RUIDOX10) apreciá que el método de suavizado de Moving Average mejoraba a las transformadas Wavelets.

La tabla 7 nos muestra el resultado sobre los conjuntos de entrenamiento más ruidosos (RUIDOCONDX10), comparando la media, mediana y sumatorio de errores cuadrados para los tres mejores sistemas: Moving Average, Wavelets Coiflet 18 y Best Located orden 14.

| | | | |
|---|---|---|---|
| RUIDOCONDX10 | | MovingAverage | 1,1998142193474 |
| | sumECM | TW Coiflet 18 | 2,71671767735478 |
| | | TW B Located 14 | 2,7177447 3165017 |
| | | MovingAverage | 0,00011998142193474 |
| | media | TW Coiflet 18 | 0,00027167176773 5478 |
| | | TW B Located 14 | 0,00027177447 3165017 |
| | | MovingAverage | 0,00011838316244483 |
| | mediana | TW Coiflet 18 | 0,00027459655755 8953 |
| | | TW B Located 14 | 0,00027476554556 7459 |

TABLA 6: Evaluación de la bondad de los sistemas de suavizado para RUIDOCONDX10

En resumen, para cerrar este apartado de experimentos iniciales, se decide adoptar, en vista de los resultados obtenidos, el suavizado de errores mediante Moving Average

### 3.2 Experimentos con Modelos Predictivos

El objetivo del presente TFM no es otro que poder determinar el mejor algoritmo predictivo de temperatura efectiva en función de los conjuntos de datos proporcionados.

Para ello, hay que tener muy encuenta que se presenta un problema de regresión en función de las

características de las variables regresoras analizadas y de la variable respuesta (telf), pero el problema siempre es el mismo, predecir o modelar la respuesta a partir de las variables regresoras.

Considerando el origen de los datos definido en el apartado 2.4, damos por supuesto que los elementos del conjunto de entrenamiento proporcionados por France Allard están correctamente clasificados sin error, de forma que los clasificadores a estudiar en el presente Trabajo son supervisados

Los clasificadores evaluados son K-Vecinos Cercanos, Máquinas de Vectores Soporte y Procesos Gausianos

Debida a la alta dimensionalidad de los datos (180 atributos) se considera oportuno estudiar la reducción de dimensionalidad a los conjuntos de datos mediante la transformación de atributos, aplicando Análisis de Componentes Principales (considerando que las componentes de alta variabilidad son las determinantes en la clase), DiffusionMaps (añadiendo un componente de aleatoriedad en la transformada) y Partial Least Square (buscando combinaciones lineales de las variables explicativas).

### 3.2.1 Clasificadores sin transformación ni reducción de atributos

En una primera aproximación sobre la predicción de la temperatura efectiva, se estudiarán los clasificadores especificados sin aplicar reducción de dimensionalidad ni transformación de datos en la etapa de preprocesado.

Para entrenamiento de los clasificadores se emplea el conjunto de datos NOMareal (conjunto de espectros NOM normalizados a areal).

El conjunto de espectros referido como RANG15 se trata del conjunto de espectros proporcionados por France Allard, filtrados por DPAC-CU2, sin discriminación entre los modelos COND y DUST.

Los conjuntos de espectros referidos a los modelos CONDareal y DUSTareal para magnitud aparente G20 se referencian como CONDG20, DUSTG20

Los conjuntos de espectros que muestran los primeros resultados de la sonda GAIA a mitad de observación se referencian como CONDG202Y y DUSTG202Y

Para validación de los clasificadores se emplean los conjuntos de espectros RANG15, CONDG20, DUSTG20, CONDG202Y y DUSTG02Y aplicando sobre ellos la normalización a areal, obteniendo los conjuntos detallados en la tabla 7.

También en la tabla 7, se detallan otros conjuntos de espectros empleados para validar los clasificadores. Estos conjuntos de espectros, etiquetados con la coletilla "movingav", se han obtenido partiendo de los anteriores conjuntos de espectros CONDG20areal, DUSTG20areal, CONDG202Yareal y DUSTG202Yareal (espectros con ruido) aplicando sobre ellos las técnicas de Moving Average.

| Conjunto de datos para validación | COND | DUST | Nº Espectros | Con ruido? |
|---|---|---|---|---|
| RANG15areal | X | X | 11000 | |
| CONDRANG20areal | X | | 10000 | SI |
| CONDRANG20arealmovingav | X | | 10000 | SI |
| DUSTRANG20areal | | X | 1000 | SI |
| DUSTRANG20arealmovingav | | X | 1000 | SI |
| CONDRANG202Yareal | X | | 10000 | SI |
| CONDRANG202Yarealmovingav | X | | 10000 | SI |
| DUSTRANG202Yareal | | X | 1000 | SI |
| DUSTRANG202Yarealmoving | | X | 1000 | SI |

Tabla 7 Conjunto de espectros empleado en la validación de los clasificadores

En los clasificadores que ofrecen la representación por medio de funciones Kernel, se realizó una experimentación inicial empleando NOMareal con validación cruzada.

El resultado de esta experimentación, determinó que tanto las máquinas vectores soporte como los procesos gausianos se comportan mejor usando un kernel basado en funciones de Base Radial Gaussiana (RBF) para obtener el nuevo espacio transformado .

La experimentación para la predicción se realizó mediante el software Weka.

A continuación se detallan las variables que se optimizaron para cada clasificador:

Para los K-vecinos cercanos::

KNN Número de vecinos cercanos a usar.

DistanceWeighting Método de ponderación de la distancia entre vecinos.

- NearestNeighbourSearchAlgorithm, Algoritmo de búsqueda del vecino más cercano.

Para las máquina de vectores soporte (SMO) fueron:

- Margen blando (variable C), al no existir una separación perfecta entre los hiperplanos, el parámetro C controla la compensación entre errores de entrenamiento y los márgenes

rigidos, creando así un margen blando (soft margin) que permite algunos errores en la clasificación a la vez que los penaliza.

— Factor Gamma (variable g) del nucleo kernel RBF.

Para los clasificadores basados en Procesos Gaussianos:

- Noise: Nos determina el nivel del Ruido Gausiano (el cual es añadido la diagonal de la Matriz de Covarianza)

- Factor Gamma (variable g) del nucleo kernel RBF

Para la optimización de todos los clasificadores y sus variables, se emplearon los conjuntos de espectros NOMarcaI para entrenar, y RANGOarcaI para validar

La búsqueda de los valores óptimos de los parámetros de cada clasificador se realizó empleando un conjunto de acciones repetitivas:

1) Asignando a las variables unos valores iniciales igual a 1.

2) Entrenamiento y validación del sistema.

3) Observación de los resultados obtenidos.

4) Comparativa de variables con mejor resultado obtenido hasta el momento

5) Escalado / modificando los valores de las variables en función de 4)

6) Retorno al punto 2)

Este bucle iterativo se repitió hasta localizar las variables del clasificador que definían un comportamiento predictivo óptimo frente a la clase Temperatura efectiva (teff) de los datos de validación

Para las aproximaciones iniciales en validación cruzada con NOMareal, se obtuvieron diferentes valores de las variables de los clasificadores que las obtenidas para los modelos predictivos entrenados con los conjuntos de entrenamiento NOMareal y validados por lo conjuntos de validación RANG15areal.

Como ya se ha comentado, para la optimización de todos los clasificadores y sus variables, se emplearon los conjuntos de espectros NOMareal para entrenar, y RANG15areal para validar.

La reevaluación de los modelos y obtención de resultados del clasificador, consistió en introducir los conjuntos de espectros con ruido (CONDG20, CONDG20movingav, DUSTG20, DUSTG20movingav, CONDG202Y, CONDG202Ymovingav, DUSTG202Y, DUSTG202Ymovingav) a la entrada de los modelos clasificadores obtenidos, sin reajuste del clasificador, y por lo tanto, manteniendo el valor de las variables obtenidas en el clasificador en las condiciones de entrenamiento con NOMareal y validación con RANG15areal.

#### 3.2.1.1 K-Vecinos Cercanos

El algoritmo de k-vecinos cercanos se implementa en Weka a través del clasificador

weka.classifiers.lazy.Ibk

En las mejores condiciones obtenidas, las variables del clasificador quedaron definidas de la siguiente forma:

KNN=6, es decir, empleando los 6 vecinos cercanos

- DistanceWeighting = "by 1/distance", es decir, usando una ponderación inversa a la distancia de cada vecino cercano.

  NearestNeighbourSearchAlgorithm ="CoverTree". La busqueda del vecino cercano emplea el algoritmo "covertree" que considera una estructura de arbol con jerarquia de

niveles, conteniendo todos los puntos del espacio métrico

La Tabla 8 muestra los resultados obtenidos tanto para validación cruzada como para la validación con RANG15area1

| | NOMarea1 | RANG15area1 |
|---|---|---|
| Correlation coefficient | 0.9999 | 0.9971 |
| Mean absolute error | 0.5319 | 29.8561 |
| Root mean squared error | 7.2932 | 40.1093 |
| Relative absolute error | 0.0922 % | 6.6002 % |
| Root relative squared error | 1.074 % | 7.3589 % |

Tabla 8 Resultados sobre KNN para validación cruzada y para el conjunto de validación sin ruido RANG15area1.

Nos apoyamos en las gráficas de dispersión sobre la validación con RANG15area1 (figuras 29 y 30) para extraer conclusiones. Téngase en cuenta que el degradado de color muestra el error en la predicción de la temperatura con respecto a la real

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

En las gráficas de dispersión de Temperatura efectiva frente a Temperatura estimada por el clasificador, la diagonal nos marca la predicción con error de desviación 0

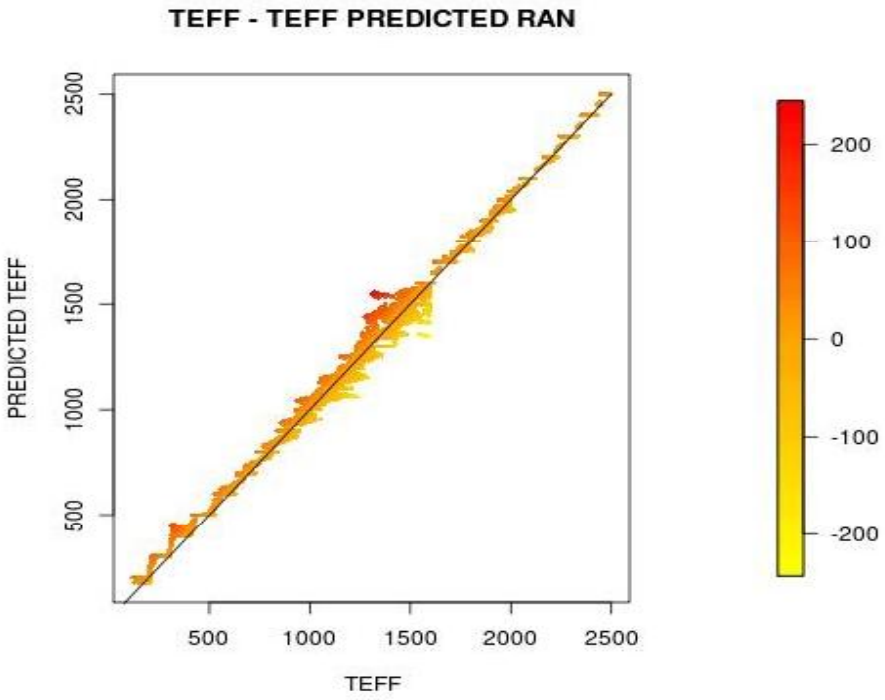


Figura 29 Gráfica de dispersión para la predicción sobre KNN de TEFF vs Teff Predicted para el conjunto de validación RANareal

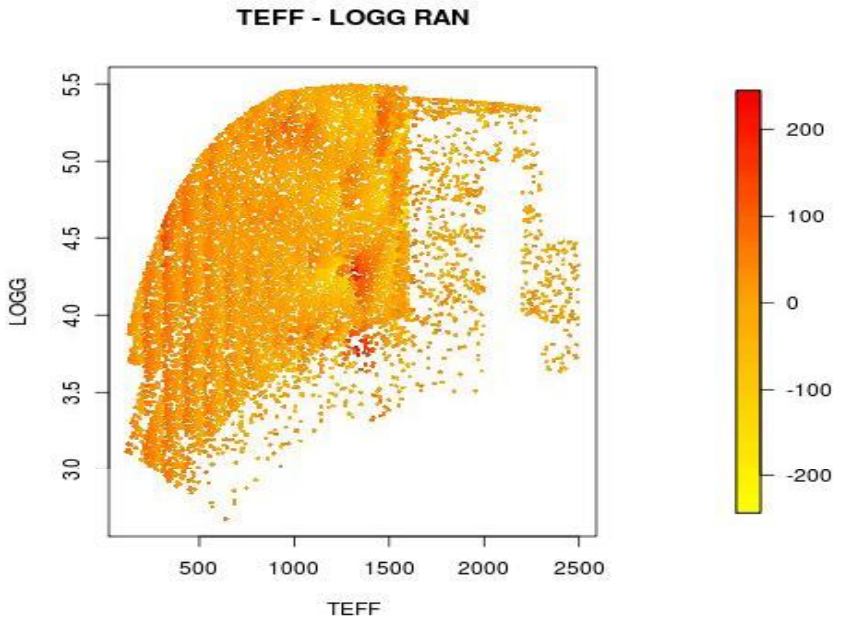


Figura 30 Gráfica de dispersión para la predicción sobre KNN de TEFF vs LOGG para el conjunto de validación RANareal

Sobre la figura 29, se observa un escalonado en los valores de temperatura por debajo de 500 grados Kelvin. Este escalonado puede coincidir con un error de interpolación en los modelos RAN generados por DPAC-CU2, sobre los modelos NOM (cuyos valores nominales de temperatura efectiva van de 100 en 100).

Para este clasificador, se observa tanto en la figura 29 como 30 que, las peores predicciones se encuentran para el rango de temperatura entre 1000 y 1500 grados Kelvin, alcanzando en este mismo rango errores de ±200 Kelvin en temperaturas efectivas muy cercanas.

Para los rangos entre 600 ° y 1000 ° Kelvin y por encima de los 1600 ° Kelvin, el clasificador tiene un comportamiento predictivo muy constante en los errores

A continuación, en un estudio más profundo se va a reevaluar el clasificador para los conjuntos de espectros para magnitud 20 comentados en la tabla 7

La tabla 9, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG20areal y CONDG20arealmoving

| MEDIDA DE ERROR | CONDG20 | CONDG20 movingav |
|---|---|---|
| Correlation coefficient | 0.9946 | 0.9936 |
| Mean absolute error | 35.0148 | 37.5764 |
| Root mean squared error | 45.1806 | 48.7934 |
| Relative absolute error | 8.1521 % | 8.7484 % |
| Root relative squared error | 8.6511 % | 9.3428 % |

Tabla 9: Resultados para KNN de los conjuntos de espectros CONDG20areal y CONDG20arealmoving

Las figuras 31 y 33 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG20areal y CONDG20arealmoving empleando el clasificador KNN.

Las figuras 32 y 34 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos

CONDG20areaI y CONDG20areaImoving empleando el clasificador KNN

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

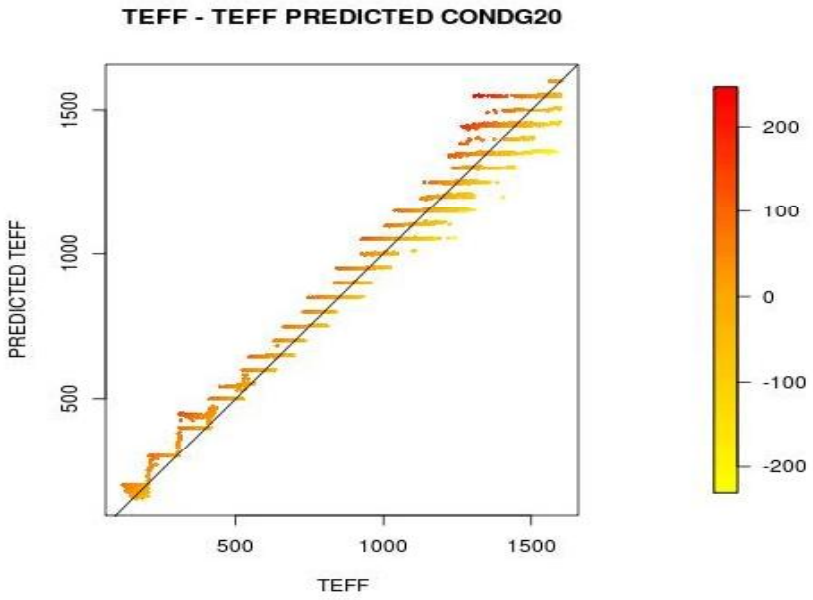


Figura 31 Gráfica de dispersión TeffI vs TeffI predicted, sobre KNN para el conjunto de espectros CONDG20areaI.

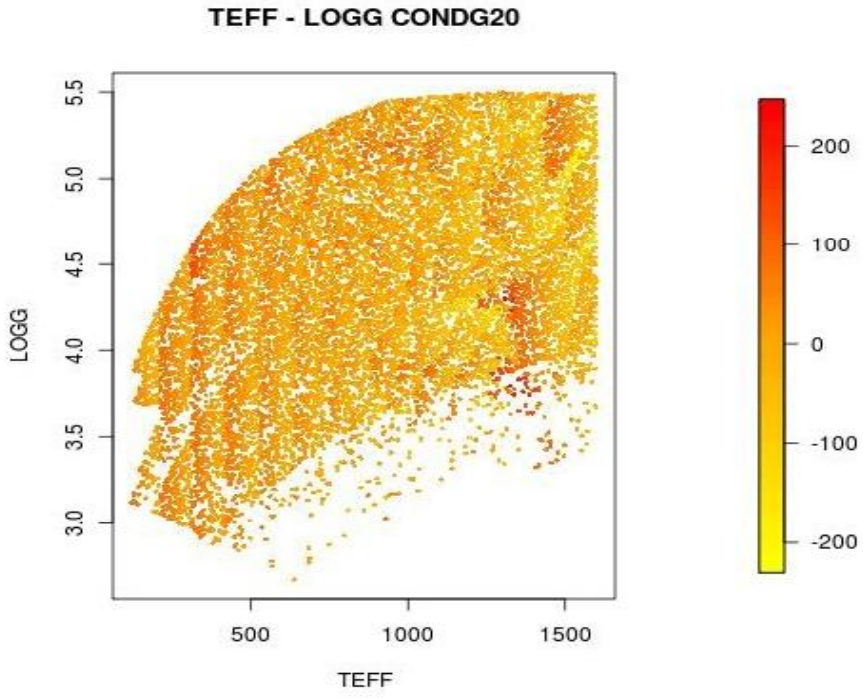

Figura 32. Gráfica de dispersión Teff vs Logg sobre KNN, para el conjunto de espectros CONDG20real

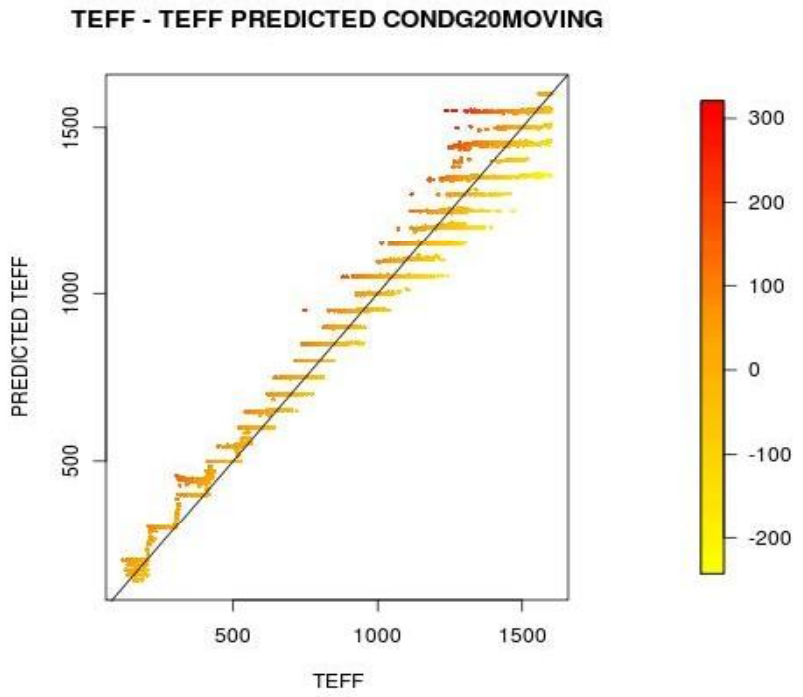


Figura 33 Gráfica de dispersión Teff vs Teff predicted, sobre KNN para el conjunto CONDG20real:moving.

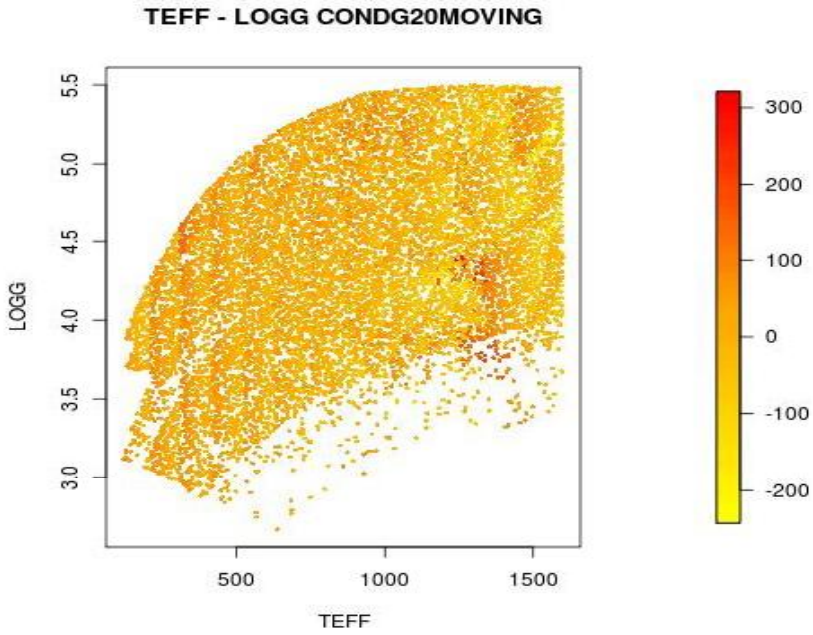

Figura 34. Gráfica de dispersión Teff vs Logg sobre KNN, para el conjunto de espectros CONDG20areal moving

Por un lado, observamos como, para el conjunto de espectros CONDG20areal, la aplicación del suavizado no solo no mejora los resultados, sino que incluso los empeora ampliando los valores de los errores máximos.

Ocurren los mismos problemas detectados para los conjuntos de datos de validación sin ruido RANG15areal:

- Escalonado en valores por debajo de 500 grados kelvin.

- Los mayores errores se concentran en el rango de temperaturas 1000 y 1500 grados kelvin

- El sistema en el grango entre 600º Kelvin y 1000 º Kelvin realiza las mejores predicciones.

A continuación, la tabla 10 nos muestra el resultado de la predicción para los conjuntos de espectros de validación para magnitud 20 DUSTG20areal y DUSTG20areal moving:

| MEDIDA DE ERROR | DUSTG20 | DUSTG20 movingav |
|---|---|---|
| Correlation coefficient | 0.9944 | 0.9923 |
| Mean absolute error | 25.5912 | 28.4631 |
| Root mean squared error | 31.7855 | 35.5959 |
| Relative absolute error | 3.7597 % | 4.1816 % |
| Root relative squared error | 4.3243 % | 4.8427 % |

Tabla 10: Resultados para KNN de los conjuntos de espectros DUSTG20areal y DUSTG20areal moving

Las figuras 35 y 37 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos DUSTG20areal y DUSTG20areal moving empleando el clasificador KNN

Las figuras 36 y 38 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos para validación DUSTG20area1 y DUSTG20area1moving empleando el clasificador KNN.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

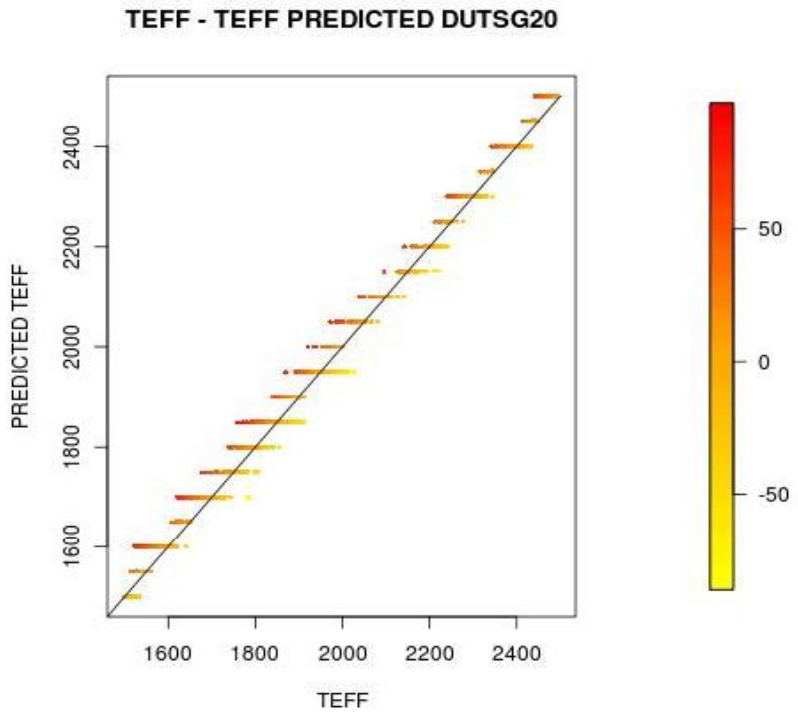


Figura 35 Grafica de dispersión TeF vs TeT predicted, sobre KNN para el conjunto de espectros DUSTG20area1.

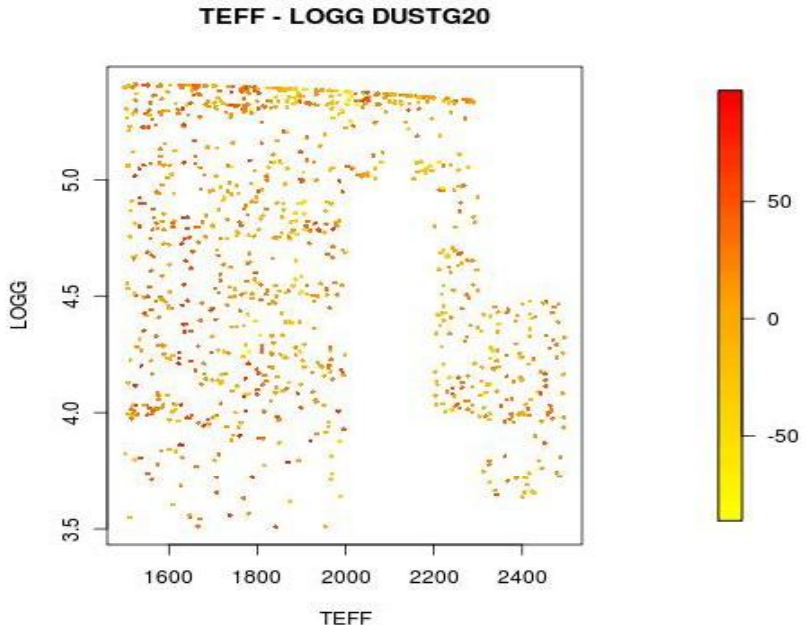


Figura 36. Gralica de dispersión Teff vs Logg sobre KNN. para el conjun.o de espectros DUSTG20area1

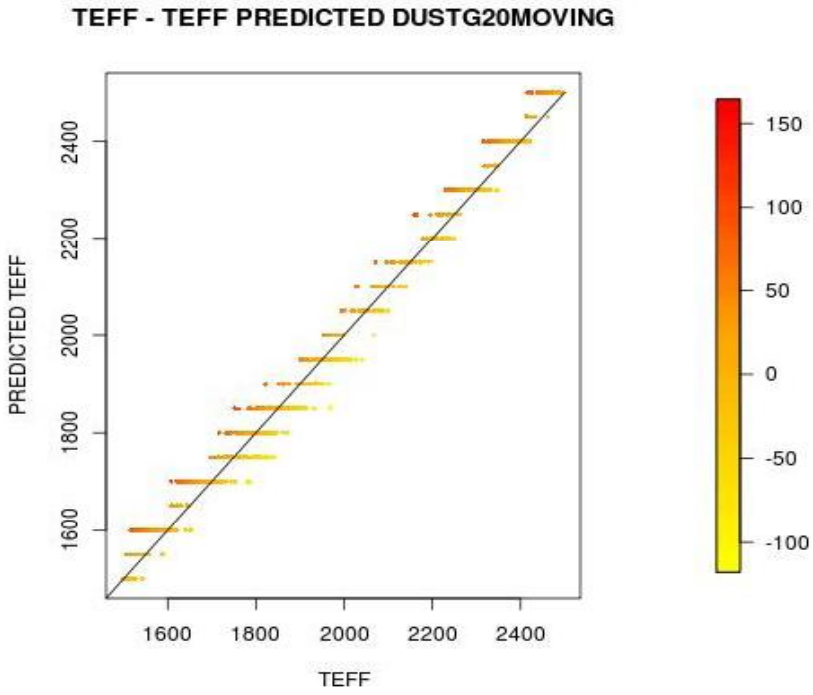


Figura 37 Gralica de dispersión Teff vs Teff predicted, sobre KNN para el conjun.o DUSTG20area1moving.

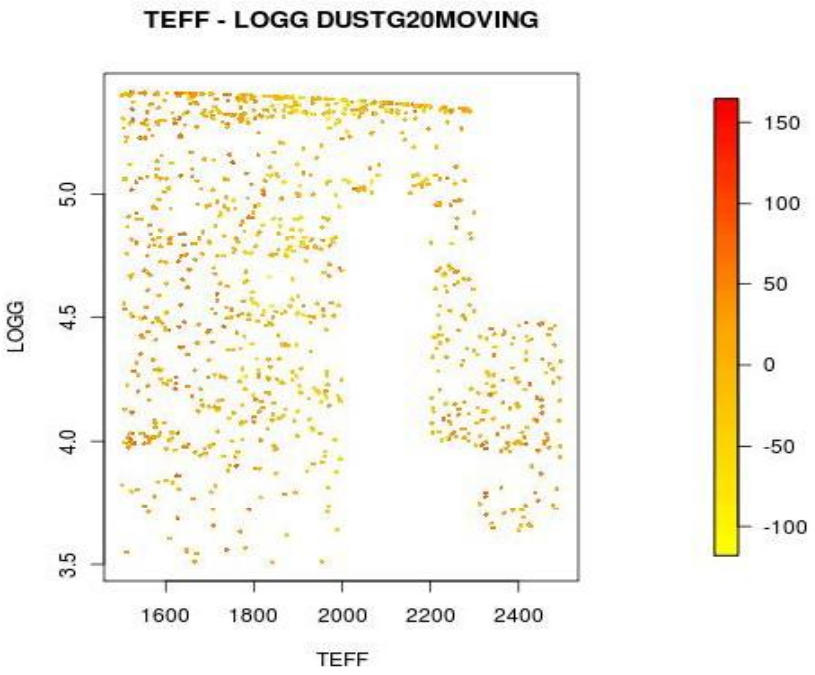


Figura 38. Gráfico de dispersión Teff vs Logg sobre KNN, para el conjunto de espectros DUSTG20area1moving

Por un lado, observamos como, para el conjunto de espectros DUSTG20area1, al igual que para los conjuntos de espectros CONDG20area1, la aplicación del suavizado no solo no mejora los resultados, sino que incluso los empeora ampliando los valores de los errores máximos.

Como los modelos DUST son espectros con temperatura efectiva superior a 1500 grados Kelvin, estos espectros no están afectados por los problemas detectados para los conjuntos de espectros de validación sin ruido RANG15area1

El sistema predictivo se comporta de una manera homogénea para todo el rango de temperaturas abarcado por los modelos DUST.

Por este motivo, el clasificador KNN ofrece mejores resultados sobre los modelos DUST. Podemos observar en las figuras 35 y 36, como los errores máximos y mínimos de predicción se reparten por igual en las diferentes temperaturas y gravedades de forma que no puede extraerse un patrón de comportamiento erroneo.

A continuación, se presenta un estudio del clasificador KNN sin reducción de dimensionalidad, para los conjuntos de validación CONDG202Y y DUSTG202Y, es decir, sobre conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación

La tabla 11, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG202Yareal y CONDG202Yarealmoving

| MEDIDA DE ERROR | CONDG202Y | CONDG202Y movingav |
|---|---|---|
| Correlation coefficient | 0.9909 | 0.989 |
| Mean absolute error | 42.2035 | 45.937 |
| Root mean squared error | 56.5447 | 62.1378 |
| Relative absolute error | 9.8257 % | 10.6949 % |
| Root relative squared error | 10.827 % | 11.898 % |

Tabla 11: Resultados para KNN de los conjuntos de espectros CONDG202Yareal

Las figuras 39 y 41 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG202Yareal y CONDG202Yarealmoving empleando el clasificador KNN

Las figuras 40 y 42 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG202Yareal y CONDG202Yarealmoving empleando el clasificador KNN.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

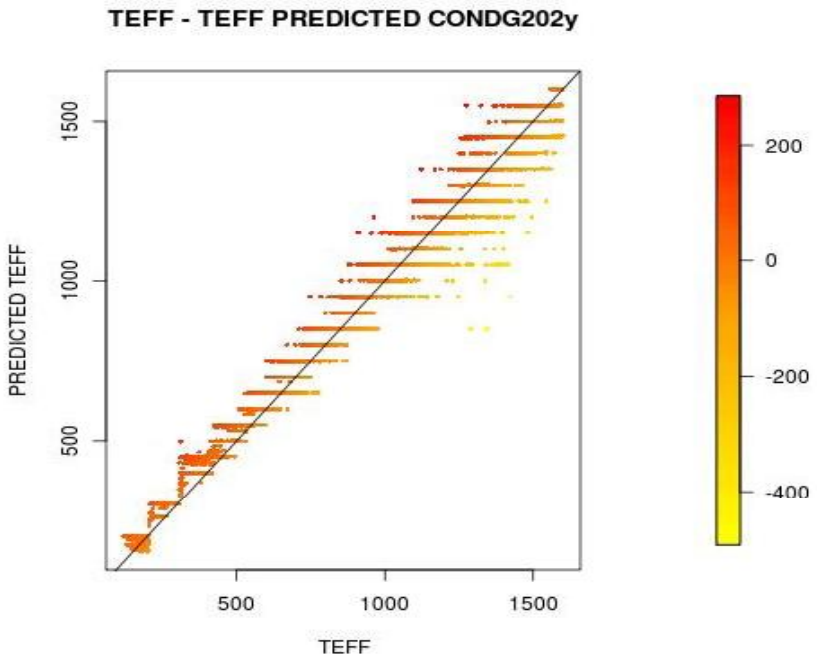


Figura 39 Gralica de dispersión TeLvs TeLpredicted, sobre KNN para el conjunto de espectros CONDG202Yarea1.

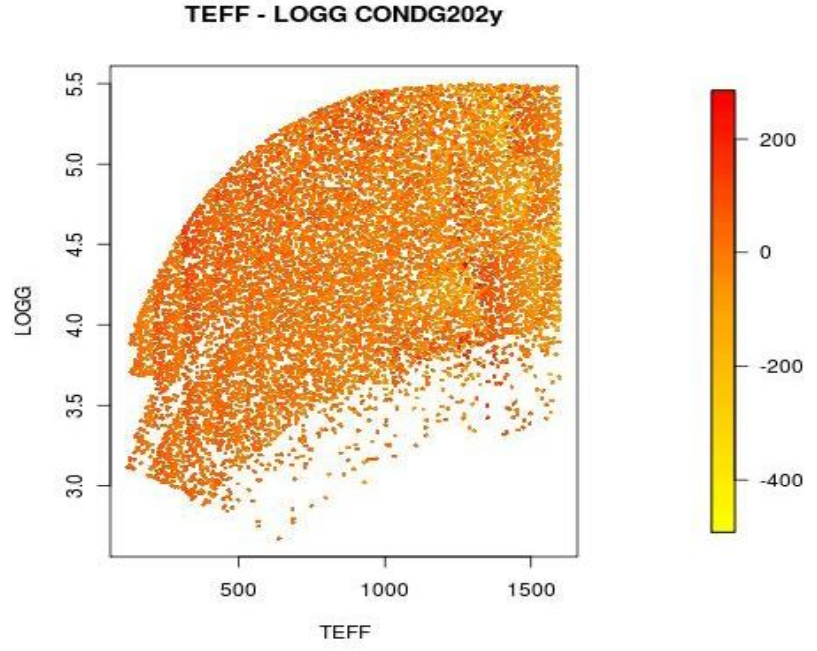


Figura 40. Gralica de dispersión TeIf vs Logg sobre KNN, para el conjunto de espectros CONDG202Yarea1

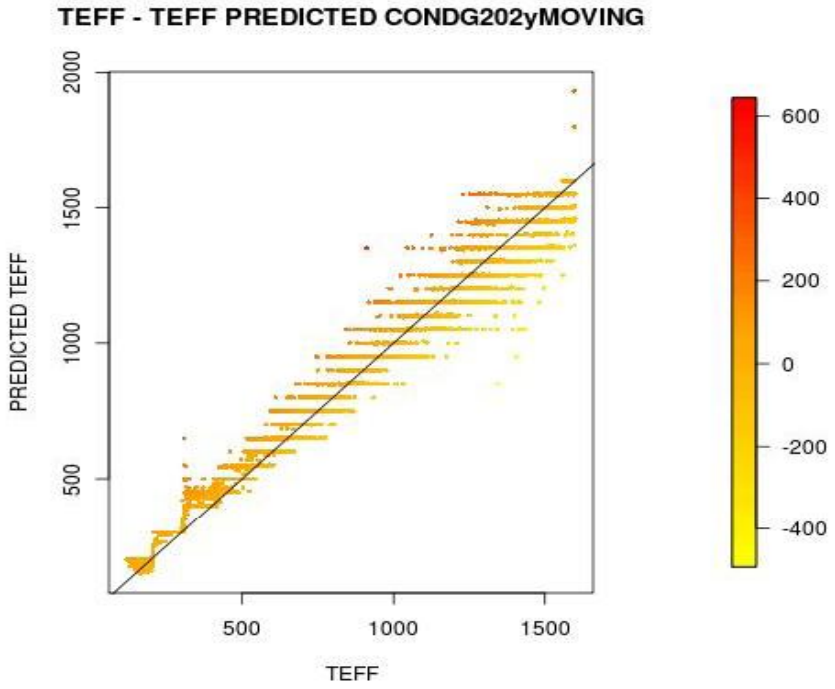


Figura 41 Grafica de dispersión Teff vs Teff predicted, sobre KNN para el conjunto CONDG202Yarea1moving.

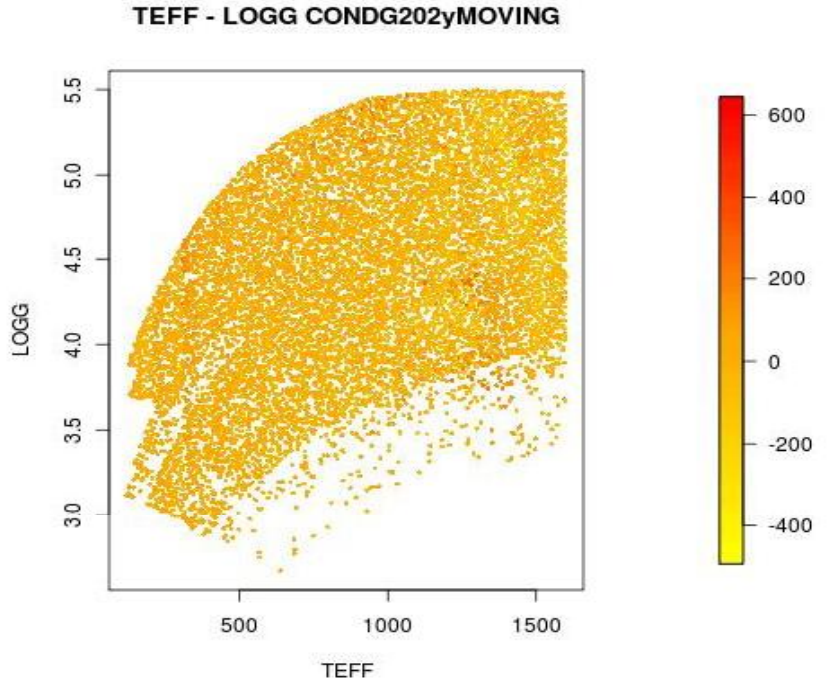


Figura 42. Grafica de dispersión Teff vs Logg sobre KNN, para el conjunto de espectros CONDG20area1moving

Se sigue observando como para el conjunto de espectros CONDG202Yarea1, la aplicación del suavizado no solo no mejora los resultados, sino que incluso los empeora ampliando los valores de los errores máximos.

Los conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación, por lo tanto, es normal que los resultados obtenidos sean peores que los conjuntos de datos de validación CONDG20area1.

Como estamos en el rango de temperaturas perteneciente a los modelos COND, se observan los mismos problemas detectados para los conjuntos de datos de validación sin ruido RANG15area1 y conjunto de datos de validación con ruido CONDG20area1:

- Escalonado en valores por debajo de 500 grados kelvin.

- Los mayores errores se concentran en el rango de temperaturas 1000° y 1500 ° kelvin.

A continuación, la tabla 12 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG202Yarea1 y DUSTG202Yarea1moving

| MEDIDA DE ERROR | DUSTG202Y | DUSTG202Y movingav |
|---|---|---|
| Correlation coefficient | 0.9772 | 0.9706 |
| Mean absolute error | 43.6111 | 49.9378 |
| Root mean squared error | 60.241 | 67.6412 |
| Relative absolute error | 6.4071 % | 7.3366 % |
| Root relative squared error | 8.1956 % | 9.2024 % |

Tabla 12: Resultados para KNN de los conjuntos de espectros DUSTG202Yarea1, DUSTG202Yarea1moving

Las figuras 43 y 45 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador KNN

Las figuras 44 y 46 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador KNN

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

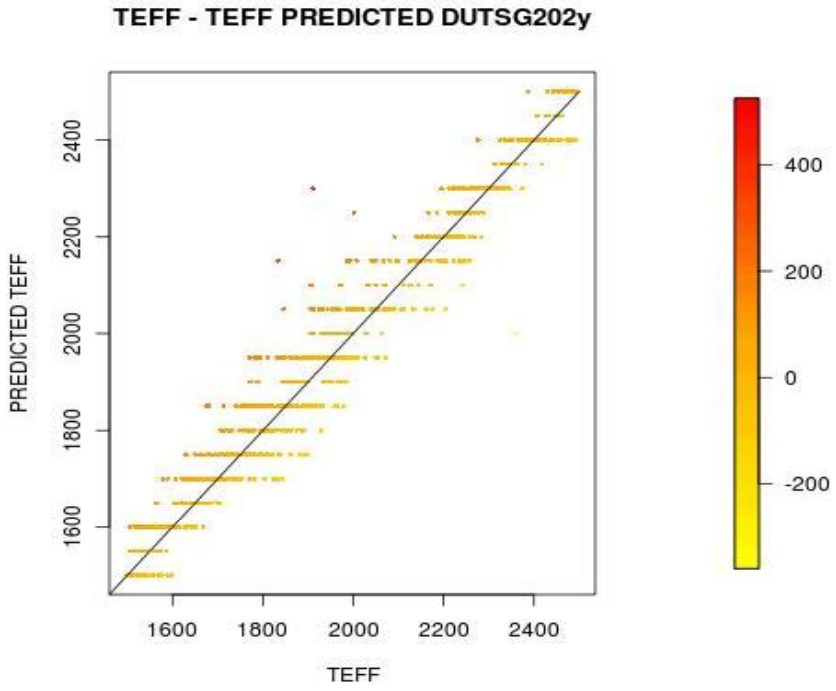


Figura 43 Gráfica de dispersión TeffTvs TeffTpredicted, sobre KNN para el conjunto de espectros DUSTG202Yarea1.

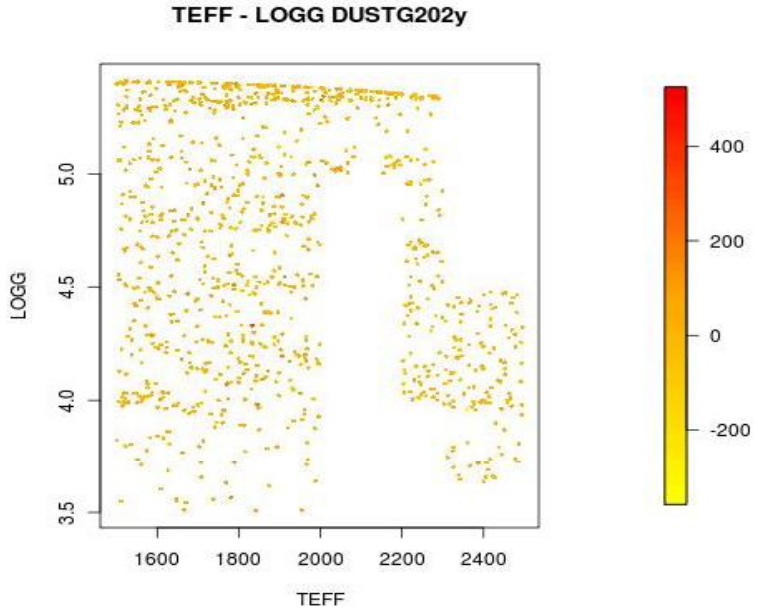


Figura 44. Gráfica de dispersión Teff vs Logg sobre KNN, para el conjunto de espectros DUSTG202Yarea1

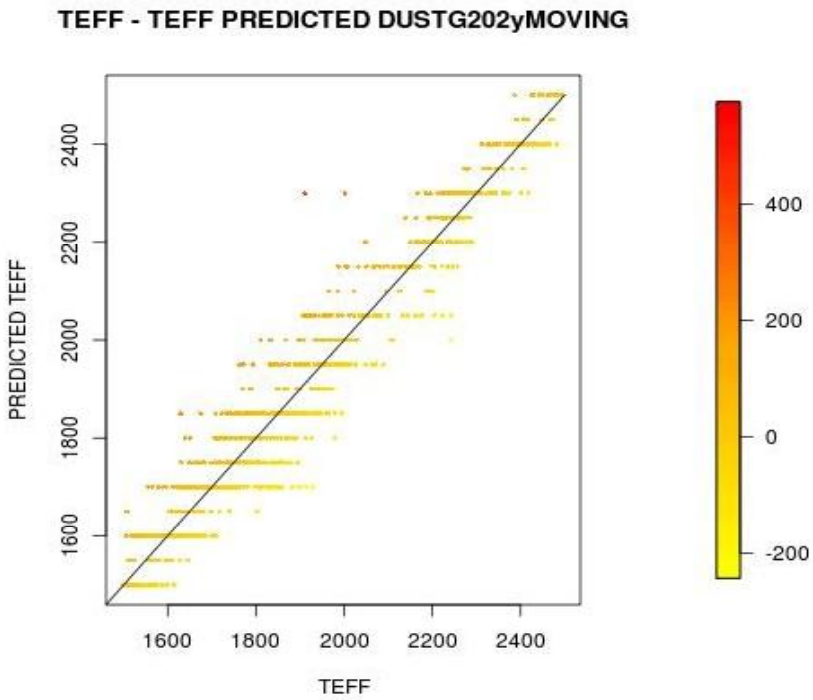


Figura 45 Gráfica de dispersión Teff vs Teff predicted, sobre KNN para el conjunto DUSTG20area1moving.

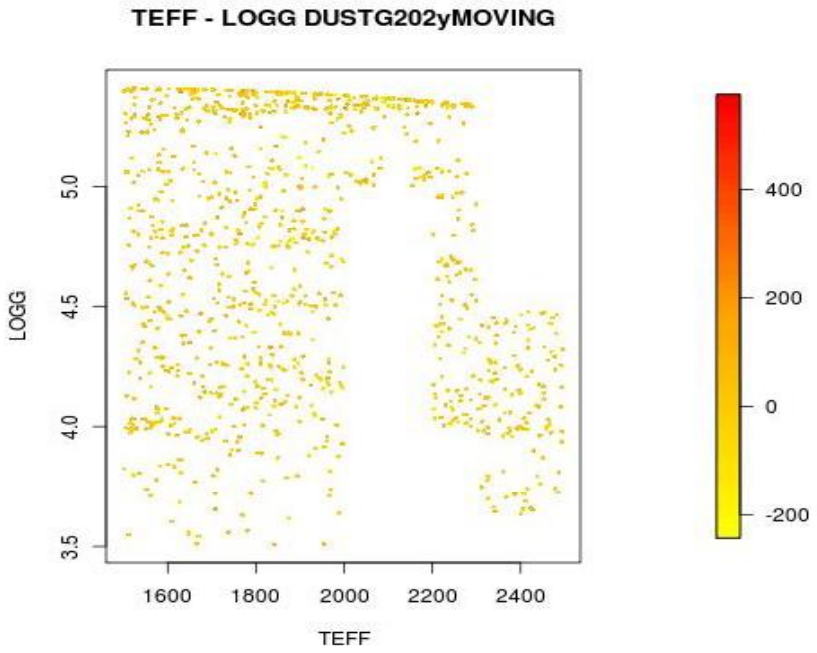


Figura 46. Gráfico de dispersión Teff vs Logg sobre KNN, para el conjunto de espectros DUSTG20area1moving

Los conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación, por lo tanto, es normal que los resultados obtenidos sean peores que los conjuntos de datos de validación DUSTG20area1

Para el conjunto de espectros DUSTG202Yarea1, al igual que para los demás conjuntos de espectros estudiados para KNN, la aplicación del suavizado no solo no mejora los resultados, sino que incluso los empeora ampliando los valores de los errores máximos

Existe una predicción realmente mala entorno a los 1900 °Kelvin que desvirtua el resultado de las predicciones, sería interesante localizar ese espectro para extraer mejores conclusiones.

Se puede observar en las figuras 43 y 44, como por lo general la predicción es más o menos homogénea y habría que estudiar al detalle los pocos ejemplos cuya predicción se encuentra con un error por encima de la media.

#### 3.2.1.2 Máquinas de vectores soporte

Para el clasificador de máquinas de vectores soporte (SMO) se empleó el algoritmo implementado en weka: weka.classifiers.functions.SMOreg, optimizando el factor gamma (g) para su kernel RBF en el valor 400 y el margen blando (c) en 2500

La Tabla 13 muestra los resultados obtenidos tanto para validación cruzada como para la validación con RANG15area1.

| | NOMarea1 | RANG15area1 |
|---|---|---|
| Correlation coefficient | 0.9999 | 0.9971 |
| Mean absolute error | 0.5319 | 29.8561 |
| Root mean squared error | 7.2932 | 40.1093 |
| Relative absolute error | 0.0922 % | 6.6002 % |
| Root relative squared error | 1.074 % | 7.3589 % |

Tabla 13: Resultados sobre SMO para validación cruzada y para el conjunto de validación sin ruido RANG15area1.

Nos apoyamos en las gráficas de dispersión sobre la validación con RANG15area1 (figuras 47 y 48) para extraer las conclusiones iniciales. Téngase en cuenta que el degradado de color muestra el error en la predicción de la temperatura con respecto a la real

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

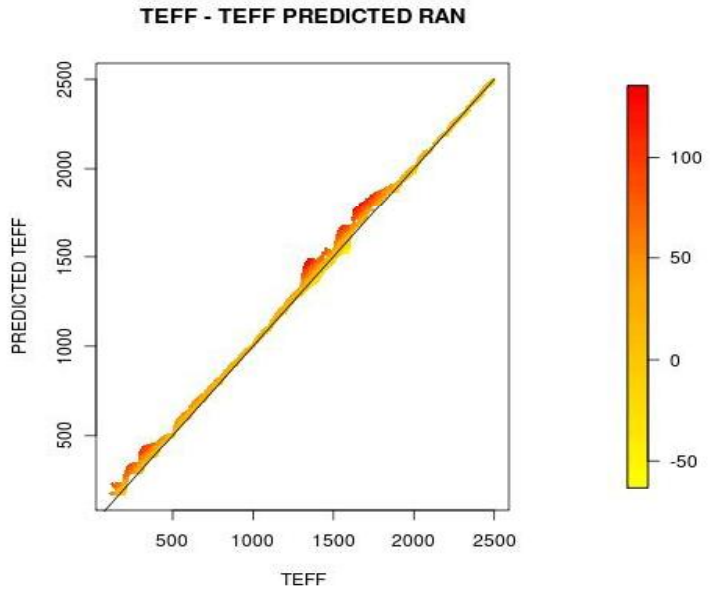


Figura 47 Gráfica de dispersión para la predicción sobre SMO de TEFF vs TEFF predicted para el conjunto de validación RANarea1

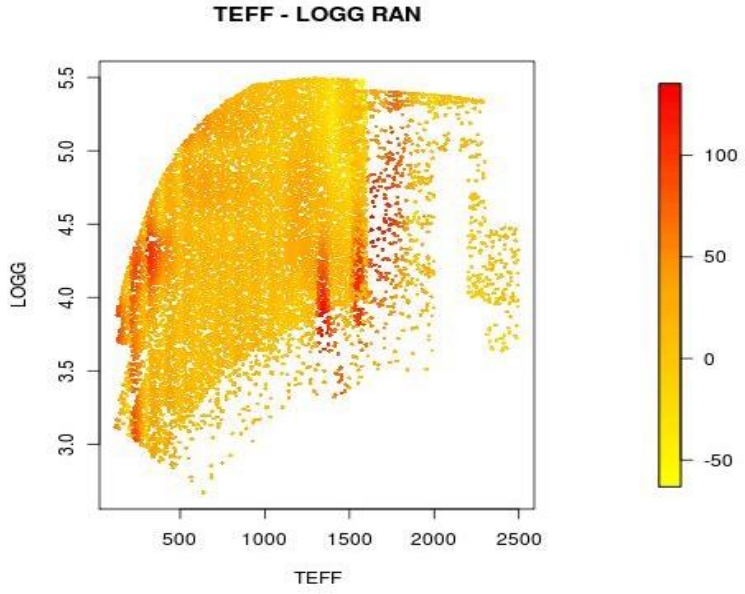


Figura 48 Gráfica de dispersión para la predicción sobre SMO de TEFF vs LOG para el conjunto de validación RANarea1.

Sobre la figura 47, se observa un escalonado en los valores de temperatura por debajo de 500 grados Kelvin. Este escalonado podría coincidir con un error de interpolación en los modelos RAN generados por DPAC-CU2, sobre los modelos NOM (cuyos valores nominales de temperatura efectiva van de 100 en 100).

Para este clasificador, se observa tanto en la figura 47 como 48 que, las mejores predicciones se encuentran para el rango de temperatura entre 500 y 1300 grados Kelvin y el rango entre 1900" y 2500" Kelvin.

Un dato a observar es que en el rango entre 500 y 1300 " Kelvin, el sistema predice generalmente temperaturas por encima de las reales

El rango de temperaturas de 1300 " a 1900 " Kelvin, el clasificador tiene las peores predicciones

A priori para el rango de temperaturas de los modelos COND el comportamiento es algo mejor que para KNN, sin embargo sería necesario realizar un estudio mediante inferencias bayesianas o un T-Student para extraer conclusiones más determinantes

A continuación, en un estudio más profundo se va a reevaluar el clasificador para los conjuntos de espectros con ruido comentados en la tabla 7.

La tabla 14, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG20area1 y CONDG20area1moving

| | CONDG20 | CONDG20 moving |
|---|---|---|
| Correlation coefficient | 0.9978 | 0.9963 |
| Mean absolute error | 24.3762 | 52.4295 |
| Root mean squared error | 32.3101 | 60.1375 |

Tabla 14: Resultados para SMO de los conjuntos de espectros CONDG20area1 y CONDG20area1moving

Las figuras 49 y 50 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG20area1 y CONDG20area1moving empleando el clasificador SMO.

Las figuras 51 y 52 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG20area1 y CONDG20area1moving empleando el clasificador SMO.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

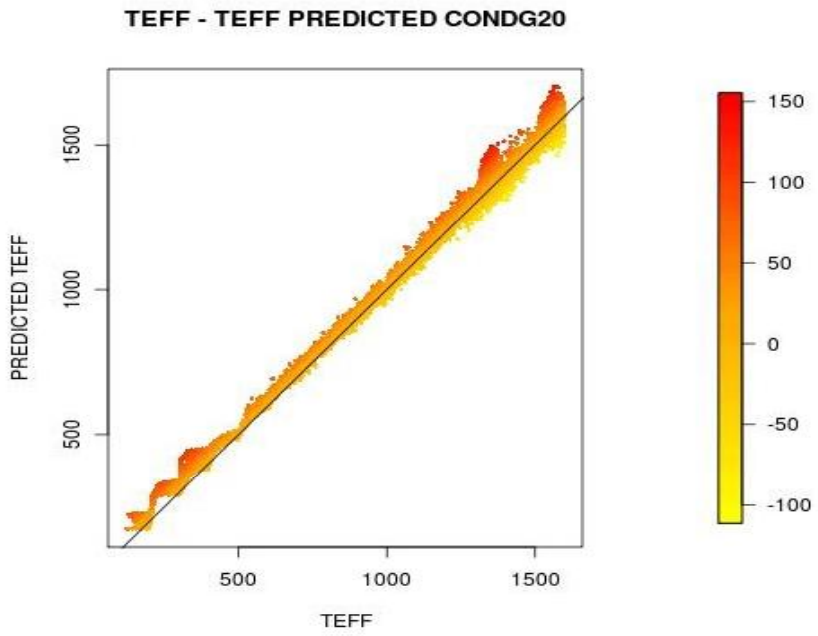


Figura 49 Grafica de dispersión Teff vs TeF predicted, sobre SMO para el conjunto de espectros CONDG20area1.

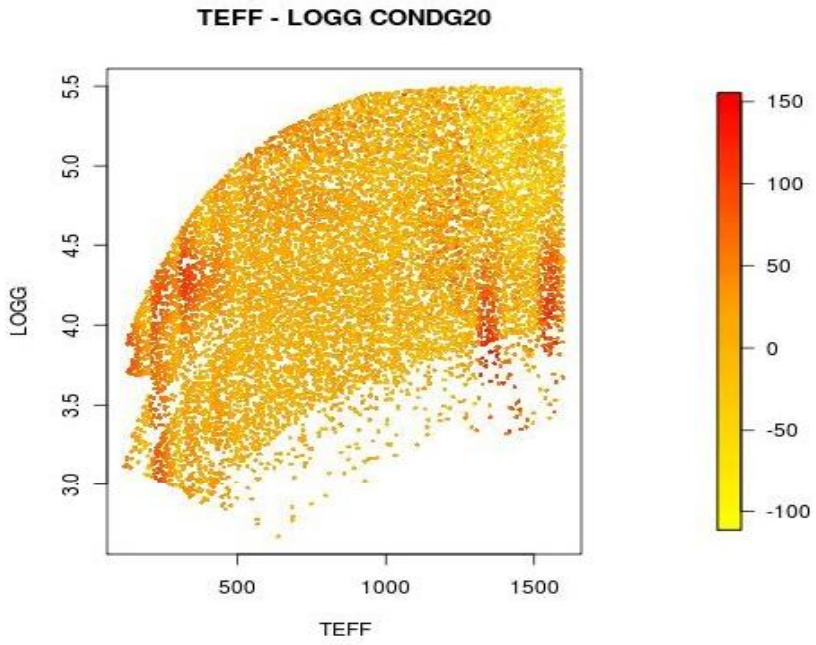


Figura 50. Grafica de dispersión Teff vs Logg sobre SMO, para el conjunto de espectros CONDG20area1

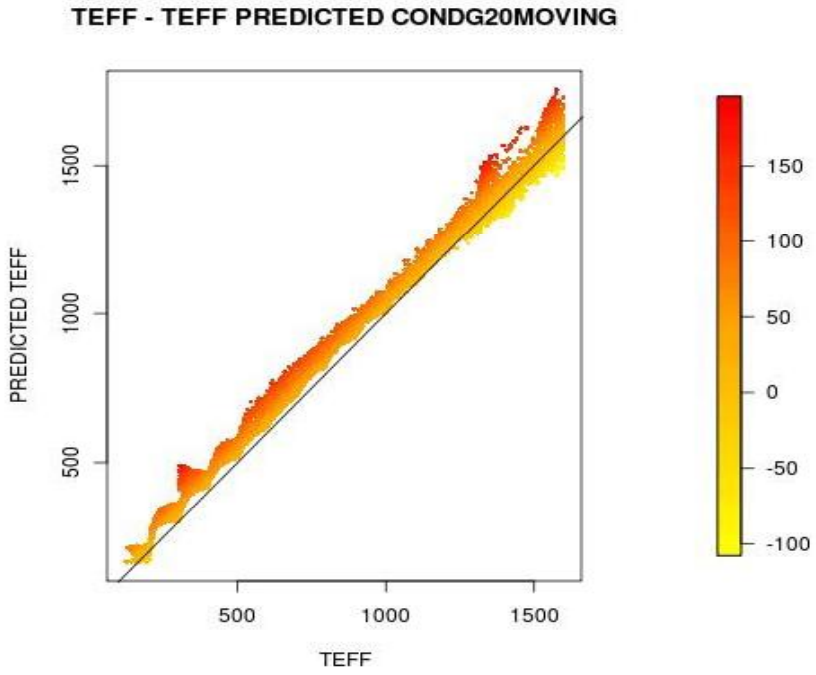


Figura 51 Grafica de dispersión Teff vs Teff predicted, sobre SMO para el conjunto CONDG20area1moving.

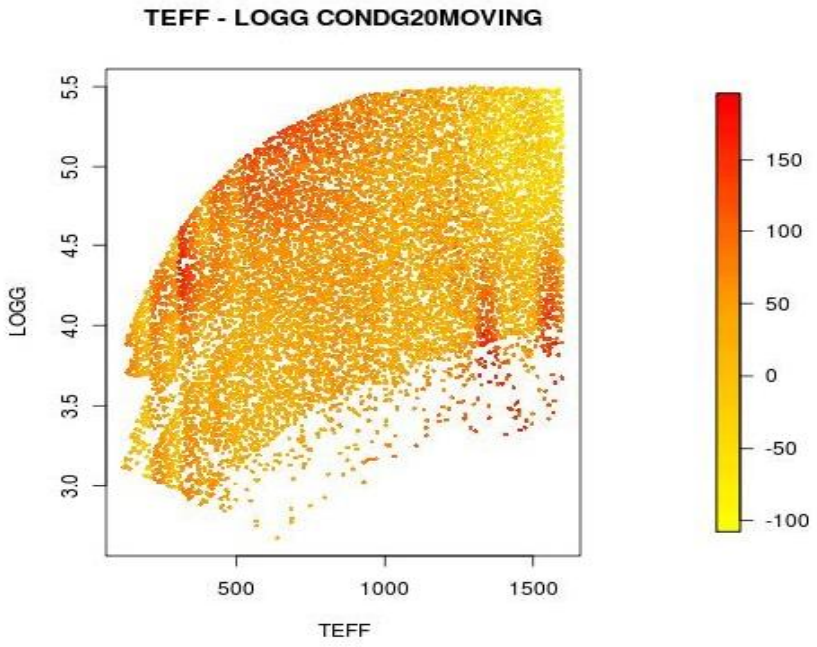


Figura 52. Gráfica de dispersión Teff vs Logg sobre SMO, para el conjunto de espectros CONDG20area1moving

Por un lado, observamos como, para el conjunto de espectros CONDG20area1, la aplicación del suavizado no solo no mejora los resultados, sino que incluso los empeora ampliando los valores de los errores máximos.

Ocurren los mismos problemas detectados para los conjuntos de datos de validación sin ruido RANG15area1:

- El escalonado en temperaturas inferiores a 500 ° Kelvin.

- Las mejores predicciones se encuentran para el rango de temperatura entre 500 y 1300 grados Kelvin

- El rango de temperaturas de 1300 ° a 1500 ° Kelvin, el clasificador tiene las peores predicciones.

A continuación, la tabla 15 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG20area1 y DUSTG20area1moving:

| | DUSTG20 | DUSTG20 moving |
|---|---|---|
| Correlation coefficient | 0.9912 | 0.9909 |
| Mean absolute error | 32.0652 | 34.1658 |
| Root mean squared error | 43.457 | 45.2301 |

Tabla 15: Resultados para SMO de los conjuntos de espectros DUSTG20area1 y DUSTG20area1moving

Las figuras 53 y 55 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos DUSTG20area1 y DUSTG20area1moving empleando el clasificador SMO

Las figuras 54 y 56 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos para validación DUSTG20area1 y DUSTG20area1moving empleando el clasificador SMO.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

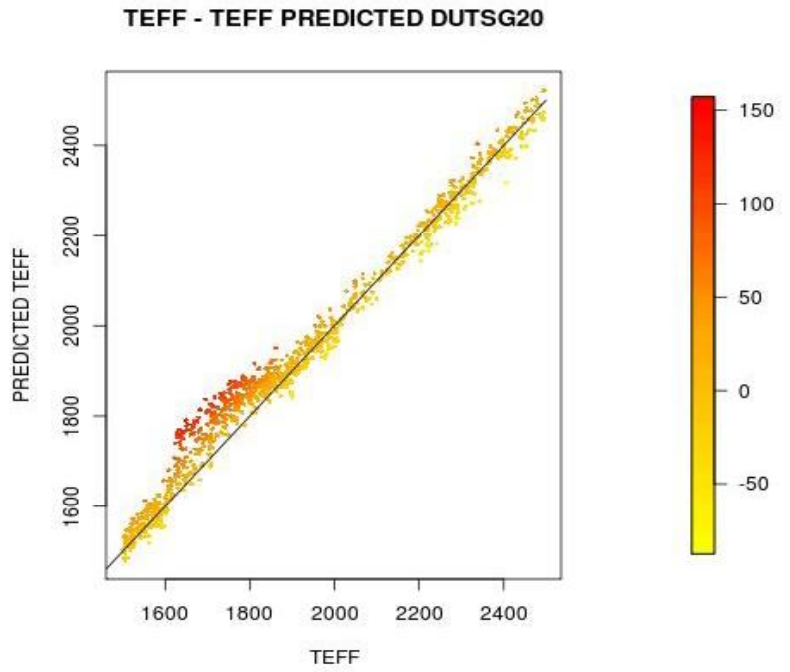


Figura 53 Grafica de dispersión TeFF vs TeFF predicted, sobre SMO para el conjunto de espectros DUSTG20area1.

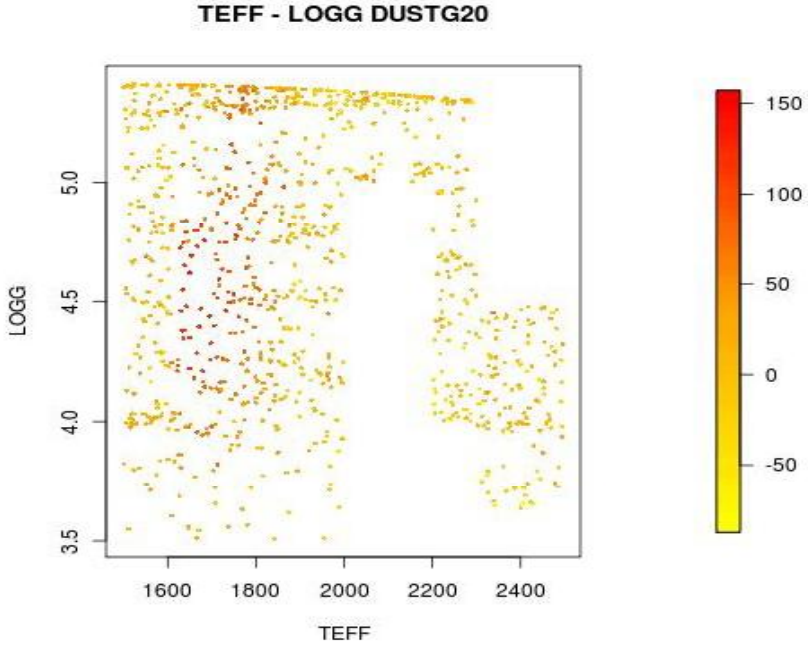


Figura 54. Grafica de dispersión TeFF vs Logg sobre SMO, para el conjunto de espectros DUSTG20area1

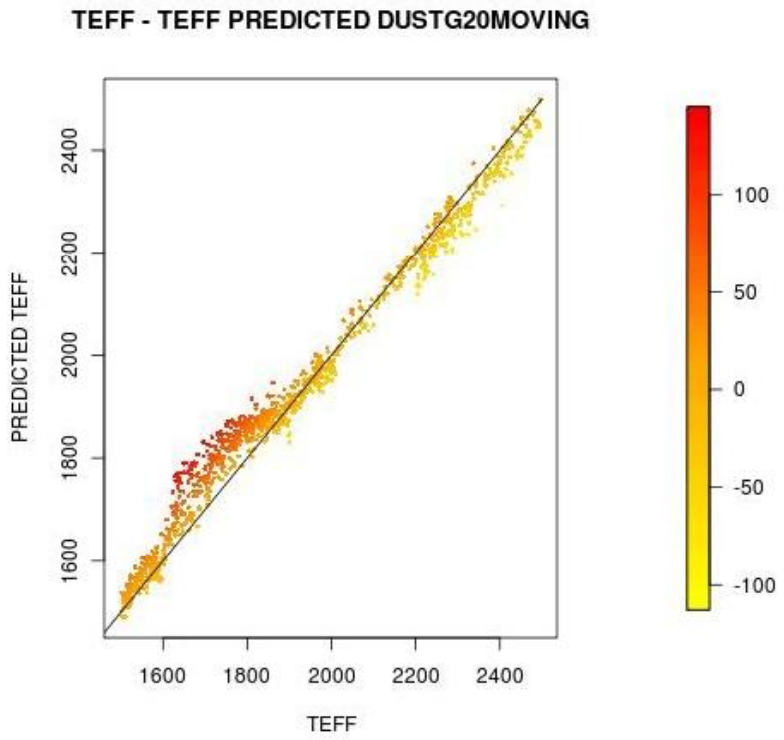


Figura 55 Gráfica de dispersión Teff vs Teff predicted, sobre SMO para el conjunto DUSTG20area1moving.

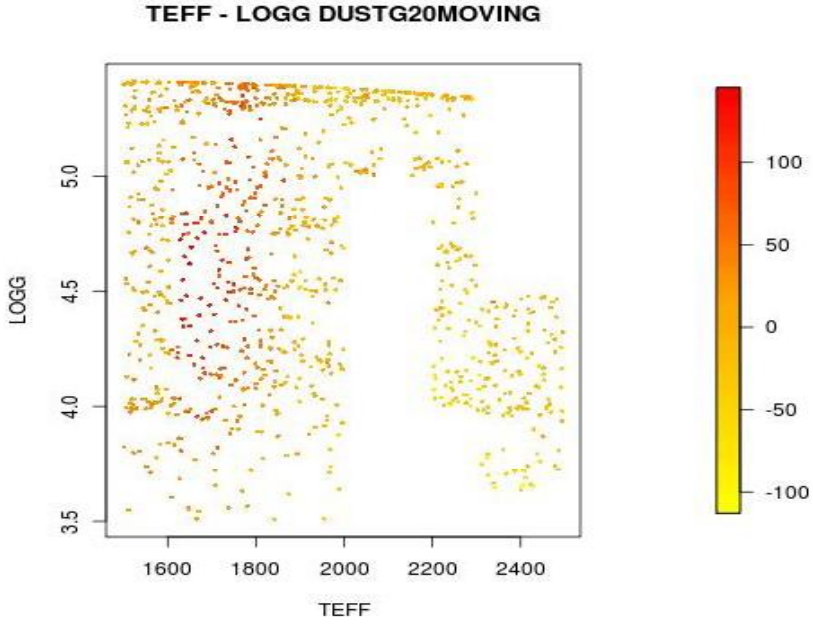


Figura 56. Gráfica de dispersión Teff vs Logg sobre SMO, para el conjunto de espectros DUSTG20area1moving

Por un lado, se observamos como para el conjunto de espectros DUSTG20areal la aplicación del suavizado no supone una gran mejora en el clasificador, salvo por el motivo que el error, de los espectros DUSTG20 por encima de 2000º Kelvin, se encuentra centrado obteniendo estimaciones predicciones por encima y por debajo del valor real, mientras que para DUSTG20movingav existe una tendencia a realizar predicciones por debajo del valor real

Por los resultados anticipados sobre RANG15areal, y teniendo en cuenta que los modelos DUST son espectros con temperatura efectiva superior a 1500 grados Kelvin, las predicciones para temperatura real entre 1500º Kelvin hasta 1900º Kelvin para valores del logaritmo de la gravedad entre 4 y 5, son las que peores estimaciones ofrecen

Por encima de los 1900º Kelvin, para los modelos DUST, las predicciones son mejores y en la mayoría de los casos, son estimaciones por debajo de la temperatura real.

A continuación, se presenta un estudio del clasificador SMO sin reducción de dimensionalidad, para los conjuntos de validación CONDG202Y y DUSTG202Y, es decir, sobre conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación

La tabla 16, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG202Yareal y CONDG202Yarealmoving

| | CONDG202Y | CONDG202Y moving |
|---|---|---|
| Correlation coefficient | 0.9788 | 0.9828 |
| Mean absolute error | 67.6749 | 66.7177 |
| Root mean squared error | 93.9842 | 83.5906 |

Tabla 16: Resultados para KNN de los conjuntos de espectros CONDG202Yareal

Las figuras 57 y 59 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos

CONDG202Yarca1 y CONDG202Yarca1moving empleando el clasificador SMO

Las figuras 58 y 60 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG202Yarca1 y CONDG202Yarca1moving empleando el clasificador KNN.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

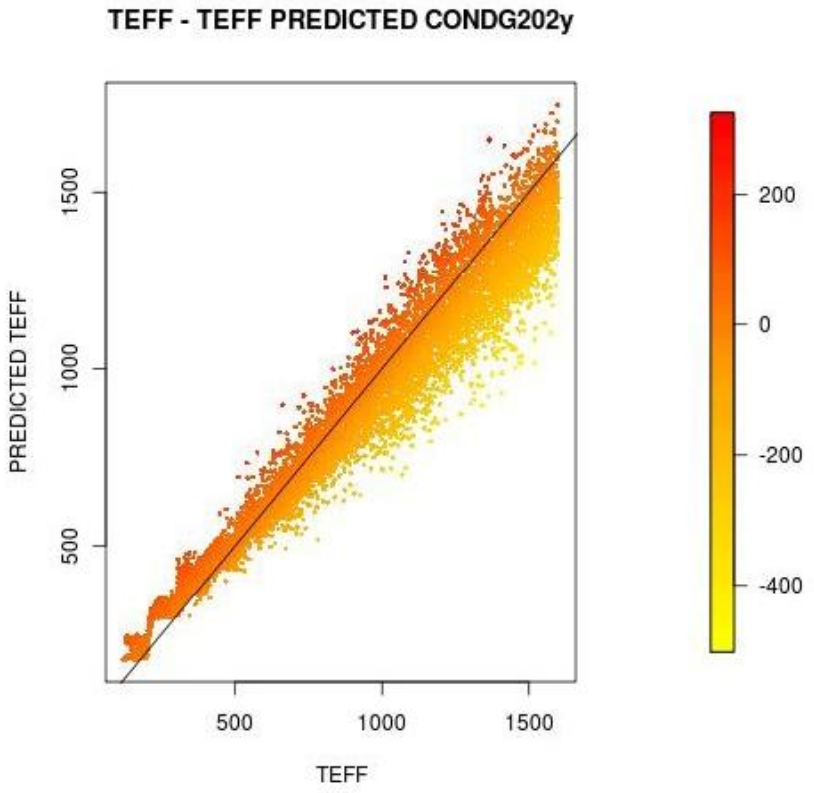


Figura 57 Gráfica de dispersión TeFF vs TeFF predicted, sobre SMO para el conjunto de espectros CONDG202Yarca1.

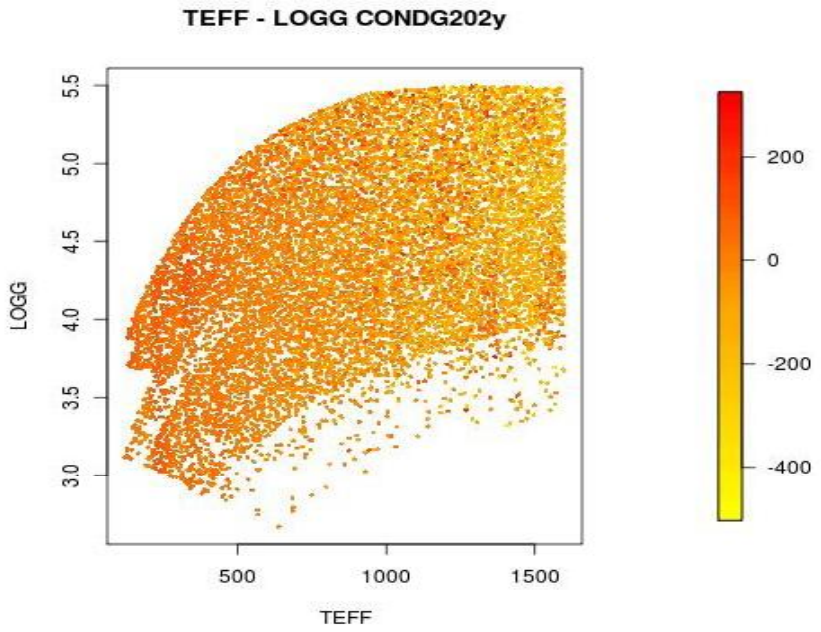


Figura 58. Gráfica de dispersión Teff vs Logg sobre SMO, para el conjunto de espectros CONDG202Yareal

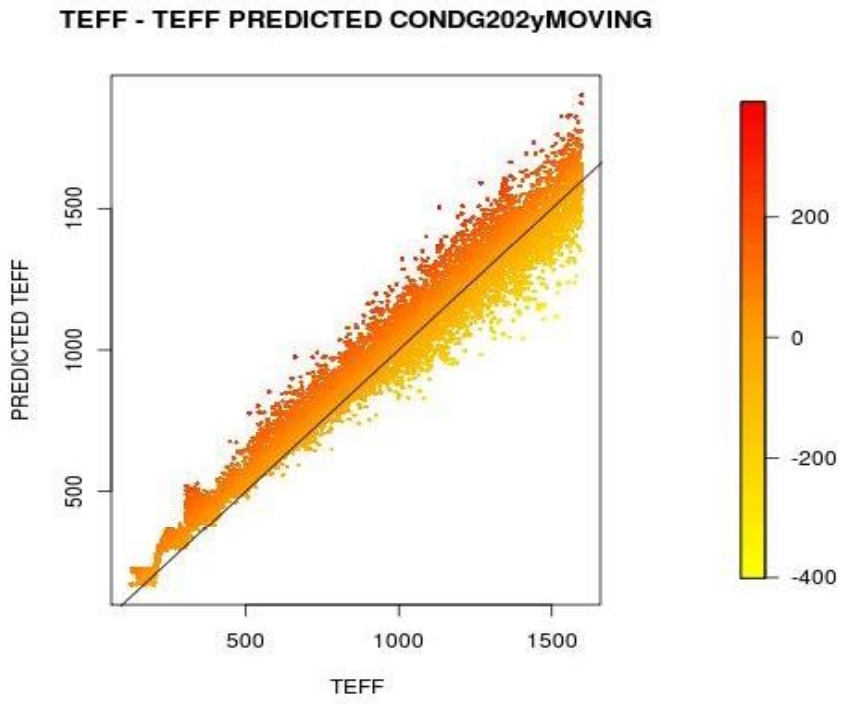


Figura 59 Gráfica de dispersión Teff vs Teff predicted, sobre SMO para el conjunto CONDG202Yarealmoving.

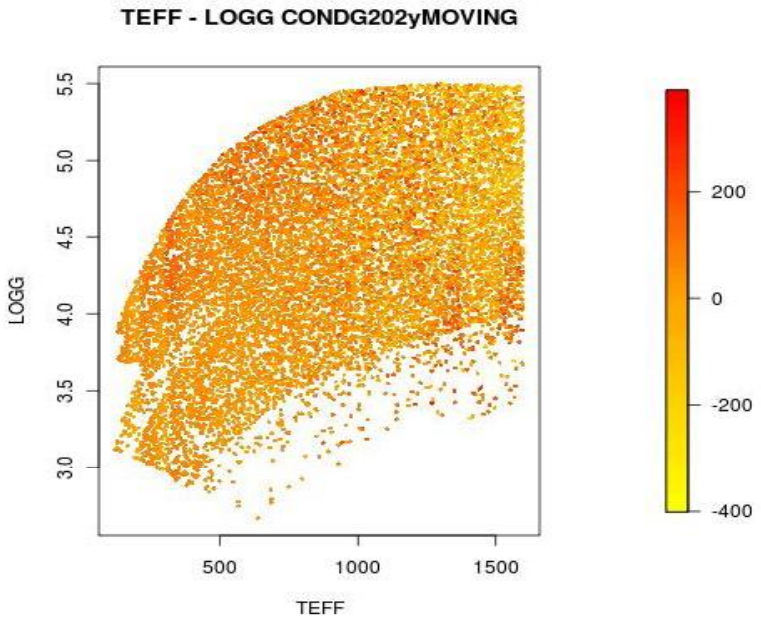


Figura 60. Gráfica de dispersión Teff vs Logg sobre SMO, para el conjunto de espectros CONDG20areal1moving

Se observa como para el conjunto de espectros CONDG202Yarea1, la aplicación del suavizado mejora los resultados, tanto en el error medio como en los valores máximos de error

Sigue observándose el escalonado sobre las temperaturas inferiores a 500 ° Kelvin

El clasificador continua ofreciendo sus mejores predicciones en el rango ente los 500 y los 1200° Kelvin, aunque se nota el aumento del error debido al ruido introducido y la falta de información que todavía la Sonda no ha recogido.

Un detalle a observar es que la mejoría del clasificador, con el uso del Moving Average sobre los conjuntos de validación, se produce entre los 500° y los 1000° Kelvin. Las predicciones en ese rango de temperatura pasan de estar por debajo de la temperatura real a estar por encima de la misma.

A continuación, la tabla 17 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG202Yarea1 y DUSTG202Yarea1moving:

| | DUSTG202Y | DUSTG202Y moving |
|---|---|---|
| Correlation coefficient | 0.9057 | 0.9295 |
| Mean absolute error | 164.1333 | 80.9071 |
| Root mean squared error | 194.304 | 103.3178 |

Tabla 17: Resultados para SMO de los conjuntos de espectros DUSTG202Yarea1, DUSTG202Yarea1moving

Las figuras 61 y 63 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador SMO

Las figuras 62 y 64 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador SMO.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

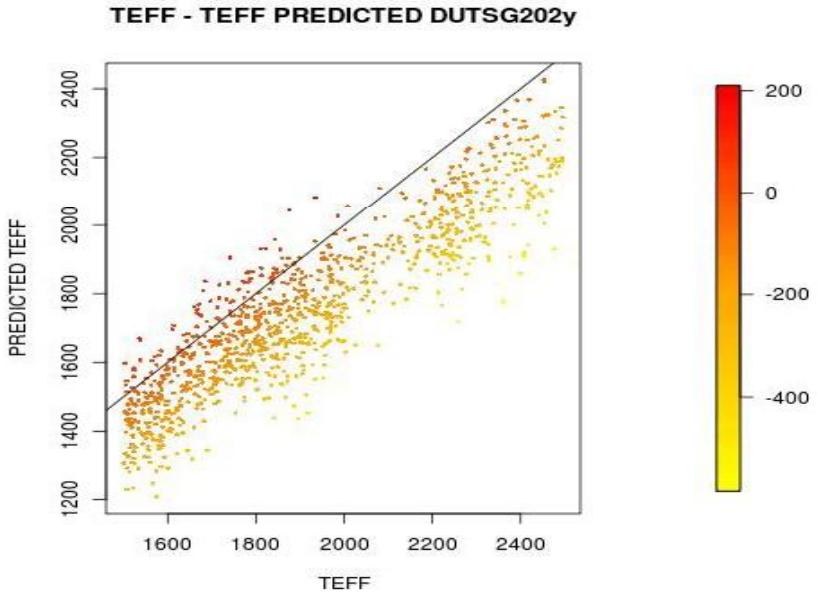


Figura 61 Grafica de dispersión TeffTvs TeffTpredicted, sobre SMO para el conjunto de espectros DUSTG202Yareal.

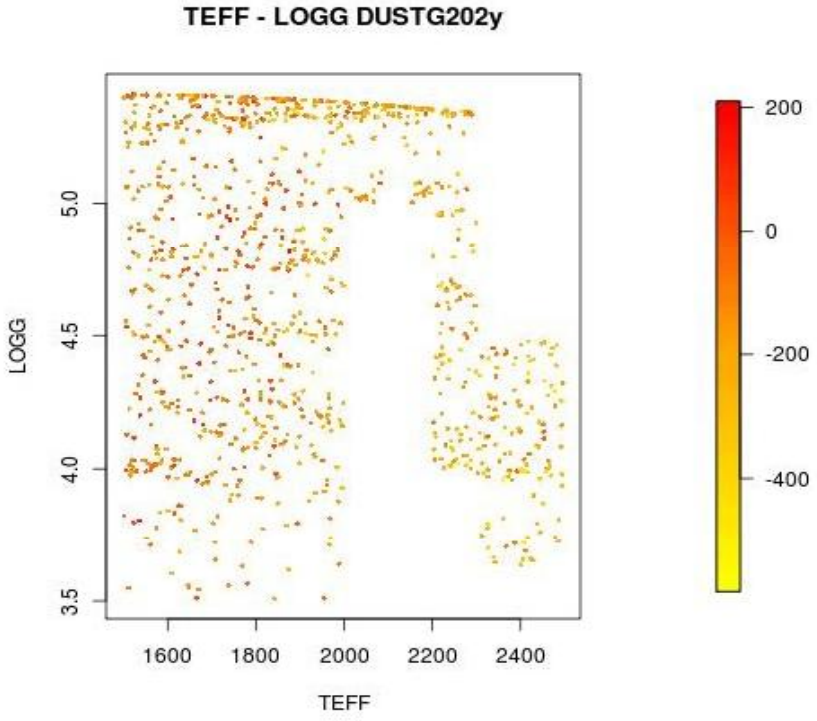


Figura 62. Grafica de dispersión Teff vs Logg sobre KNN, para el conjunto de espectros DUSTG202Yareal

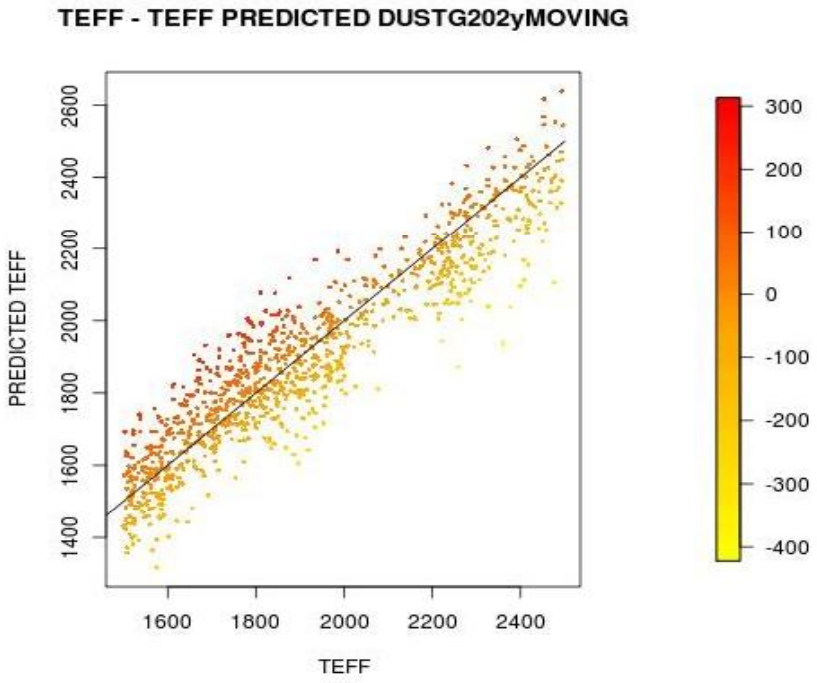


Figura 63 Gráfica de dispersión Teff vs Teff predicted, sobre SMO para el conjunto DUSTG20area1moving.

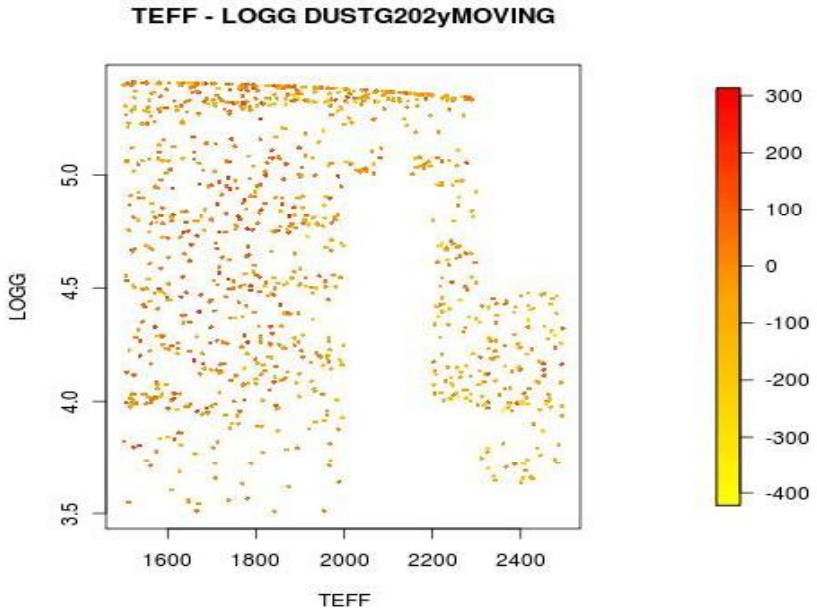


Figura 64. Gráfica de dispersión Teff vs Logg sobre SMO, para el conjunto de espectros DUSTG20area1moving

Los conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación, por lo tanto, es normal que los resultados obtenidos sean peores que los conjuntos de datos de validación DUSTG20area1

Para el conjunto de espectros DUSTG202Yarea1, al igual que para el conjunto de espectros estudiados de CONDG202Yarea1 para SMO, la aplicación del suavizado mejora los resultados en el error cuadrático medio debido a que:

- Mientras que para el conjunto de datos de validación DUSTG202Yarea1, la mayoría de las predicciones se encontraban por debajo de la temperatura real (tal y como se observa en la figura 61), para el conjunto de datos DUSTG202Yarea1 con suavizado MovingAverage el error en las predicciones (figura 63) queda repartido de una forma más o menos proprocional en predicciones por encima y predicciones por debajo de la real.

Observando las figuras 61, 62, 63 y 64 no se pueden extraer conclusiones que nos determinen algún rango de temperaturas sobre las cuales las predicciones sean mejores

#### 3.2.1.3 Procesos Gausianos

La ejecución del clasificador Procesos Gausianos en Weka se realiza a través de la función: weka.classifiers.functions.GaussianProcesses

Se optimizaron las variables comentadas en el apartado 3.2.1, el factor gamma(g) para el kernel RBF para un valor de 1.34 y la variable ruido(noise) con un valor de 0.04

La Tabla 18 muestra los resultados obtenidos para los valores comentados, tanto para validación cruzada como para la validación con RANG15area1

| | NOMarea1 | RANarea1 |
|---|---|---|
| Correlation coefficient | 0.9996 | 0.9993 |
| Mean absolute error | 8.266 | 18.082 |
| Root mean squared error | 18.72 | 24.0062 |
| Relative absolute error | 1.4329 % | 3.9973 % |
| Root relative squared error | 2.7568 % | 4.4045 % |

Tabla 18: Resultados sobre GPs para validación cruzada y para el conjunto de validación sin ruido RANG15area1.

Nos apoyamos en las gráficas de dispersión sobre la validación con RANG15area1 (figuras 65 y 66) para extraer conclusiones. Téngase en cuenta que el degradado de color muestra el error en la predicción de la temperatura con respecto a la real.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

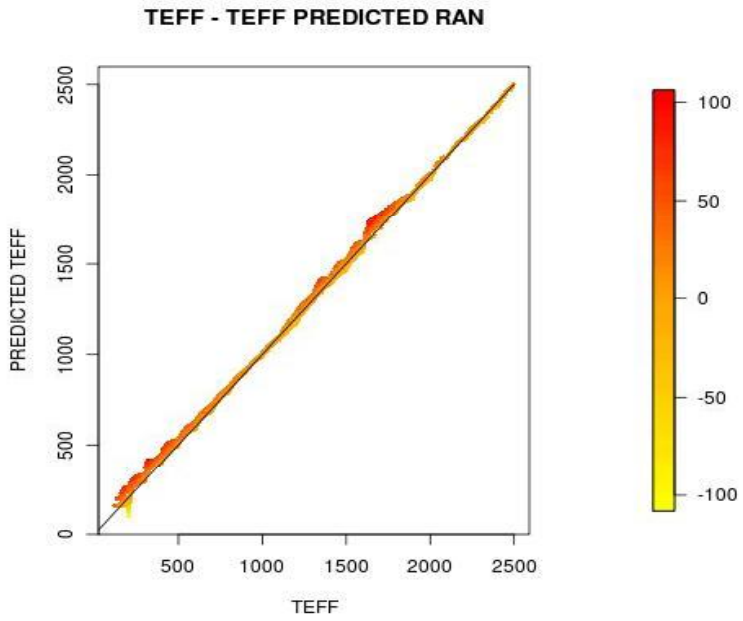


Figura 65 Grafica de dispersión para la predicción sobre GPs de TEFF vs TEFF predicted para el conjunto de validacion RANarea1

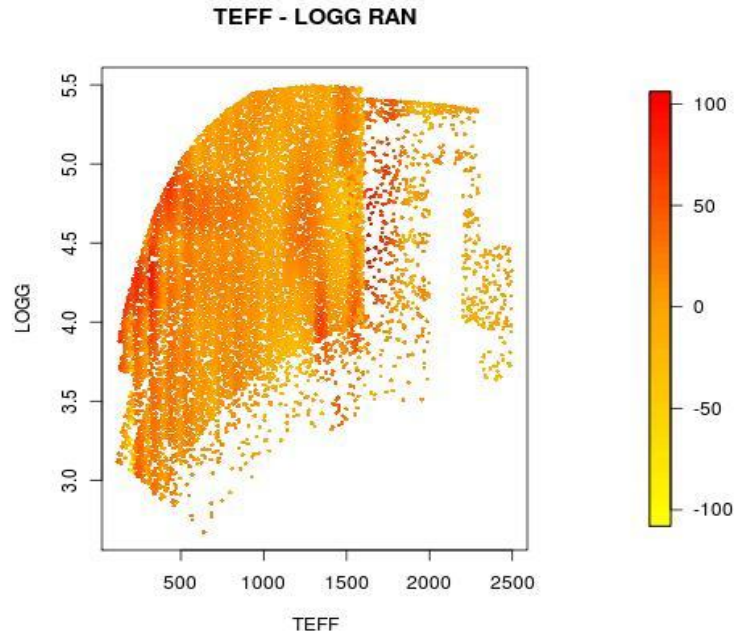


Figura 66 Grafica de dispersión para la predicción sobre GPs de TEFF vs LOG para el conjunto de validación RANarea1.

Sobre la figura 65, se observa que el escalonado existente en los otros clasificadores, también es latente par los valores de temperatura por debajo de 500 grados Kelvin  Sin embargo, es mucho menos visible que en los otros clasificadores observados hasta el momento

Para este clasificador, se observa tanto en la figura 65 como 66 que, al igual que las Máquinas de Vectores Soporte, las mejores predicciones se encuentran para el rango de temperatura entre 500 y 1300 grados Kelvin y el rango entre 1900" y 2500 " Kelvin.

El rango de temperaturas de 1300 " a 1900 " Kelvin, el clasificador tiene las peores predicciones.

A continuación, en un estudio más profundo se va a reevaluar el clasificador para los conjuntos de espectros con ruido comentados en la tabla 7

La tabla 19, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG20area1 y  CONDG20area1moving

| | CONDG20 | CONDG20 moving |
|---|---|---|
| Correlation coefficient | 0.9973 | 0.9973 |
| Mean absolute error | 26.6147 | 33.9793 |
| Root mean squared error | 33.3775 | 41.2153 |

Tabla 19: Resultados para GPs de los conjuntos de espectros CONDG20area1 y CONDG20area1moving

Las figuras 67 y 69 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG20area1 y CONDG20area1moving empleando el clasificador GPs.

Las figuras 68 y 70 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG20area1 y CONDG20area1moving empleando el clasificadorGPs.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

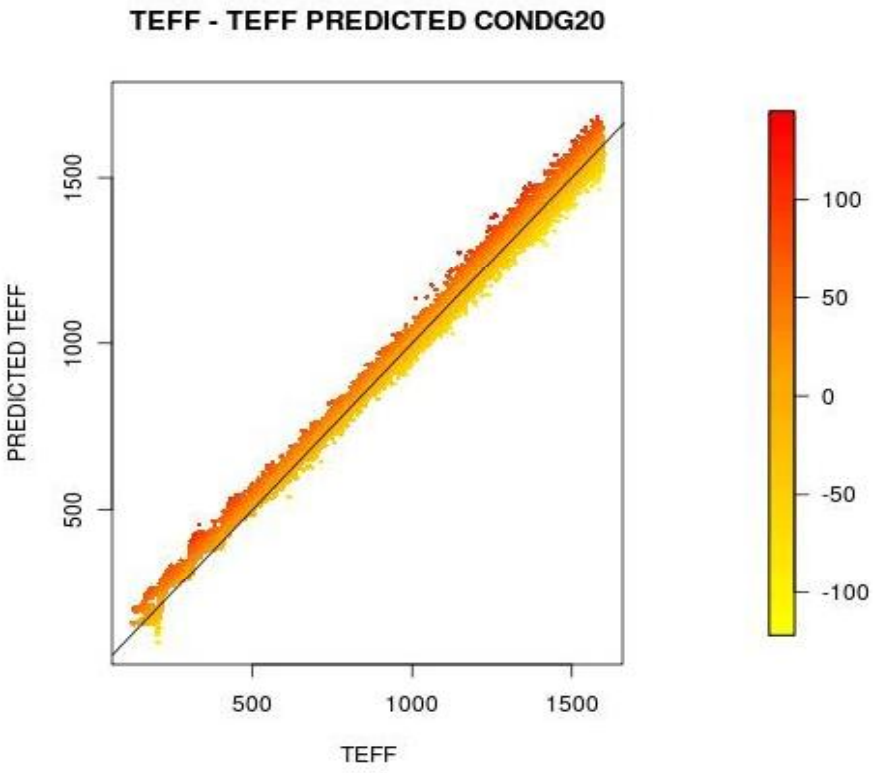


Figura 67 Gráfica de dispersión Teff vs Teff predicted, sobre GPs para el conjunto de espectros CONDG20area1.

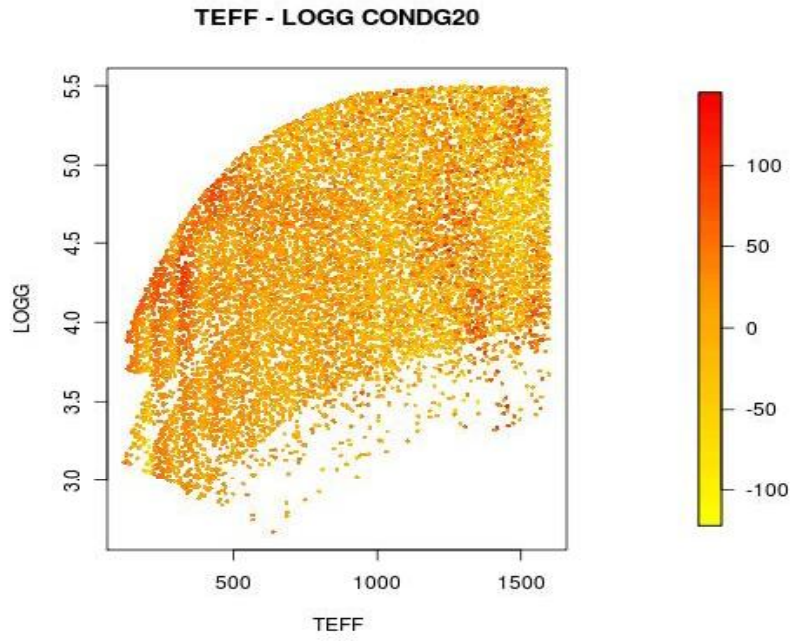


Figura 68. Gráfica de dispersión TeIf vs Logg sobre GPs, para el conjunto de espectros CONDG20areal

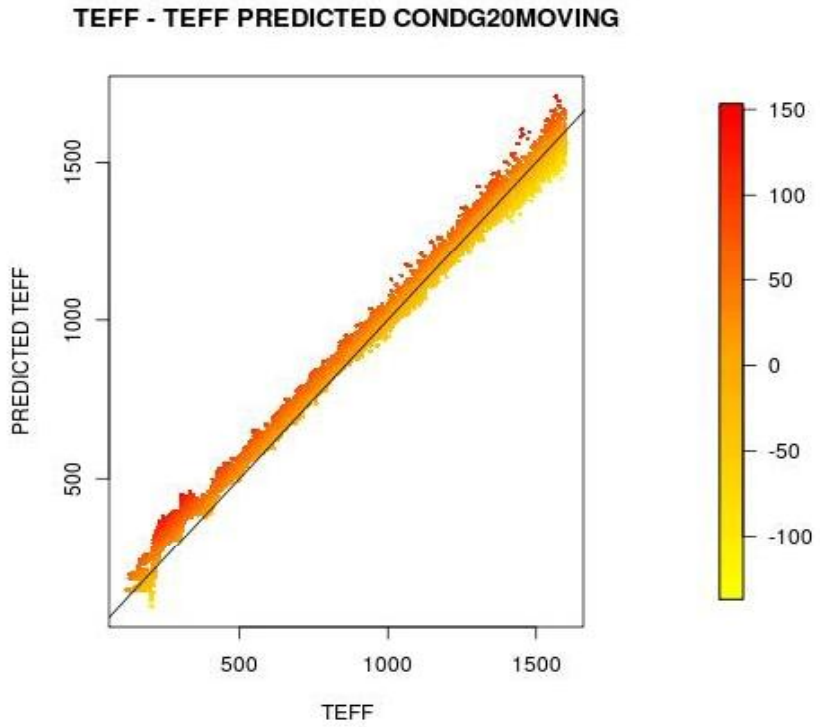


Figura 69 Gráfica de dispersión TeIf vs TeIf predicted, sobre GPs para el conjunto CONDG20areal:moving.

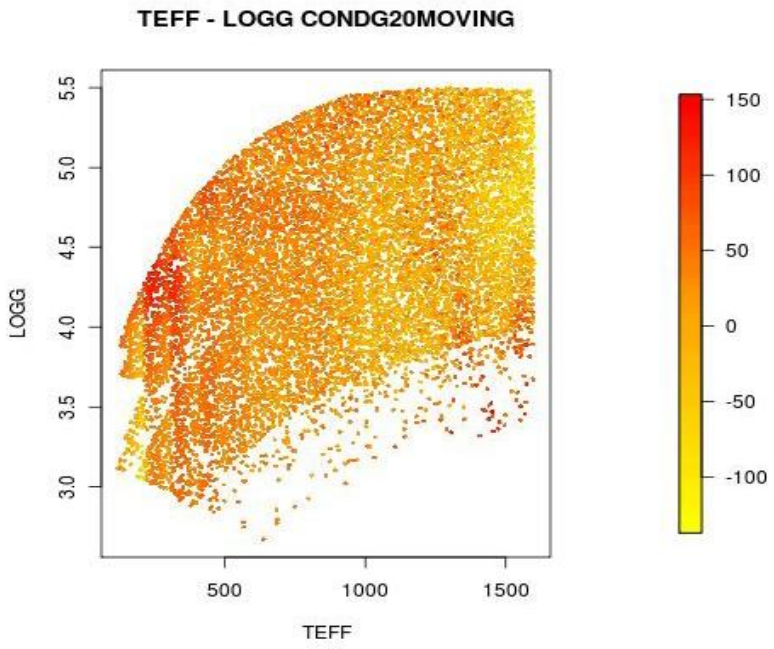


Figura 70. Gráfico de dispersión Teff vs Logg sobre GPs, para el conjunto de espectros CONDG20area1moving

Por un lado, observamos como, para el conjunto de espectros CONDG20area1, la aplicación del suavizado no solo no mejora los resultados, sino que incluso los empeora ampliando los valores de los errores máximos.

Ocurren los mismos problemas detectados para los conjuntos de datos de validación sin ruido RANG15area1:

- El escalonado en temperaturas inferiores a 500 ° Kelvin.

- Las mejores predicciones se encuentran en el rango de temperatura entre 500º y 1300ºK.

- Las predicciones entre 500º y 900º K se encuentran por debajo de la temperatura real.

- El rango de temperaturas de 1300º a 1500º K, el clasificador tiene las peores predicciones

A continuación, la tabla 20 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG20area1 y DUSTG20area1moving:

| | DUSTG20 | DUSTG20 moving |
|---|---|---|
| Correlation coefficient | 0.9885 | 0.9895 |
| Mean absolute error | 34.8482 | 36.1675 |
| Root mean squared error | 44.5842 | 46.2995 |

Tabla 20: Resultados para GPs de los conjuntos de espectros DUSTG20area1 y DUSTG20area1moving

Las figuras 71 y 73 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos DUSTG20area1 y DUSTG20area1moving empleando el clasificador GPs.

Las figuras 72 y 74 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos para validación DUSTG20area1 y DUSTG20area1moving empleando el clasificador GPs.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

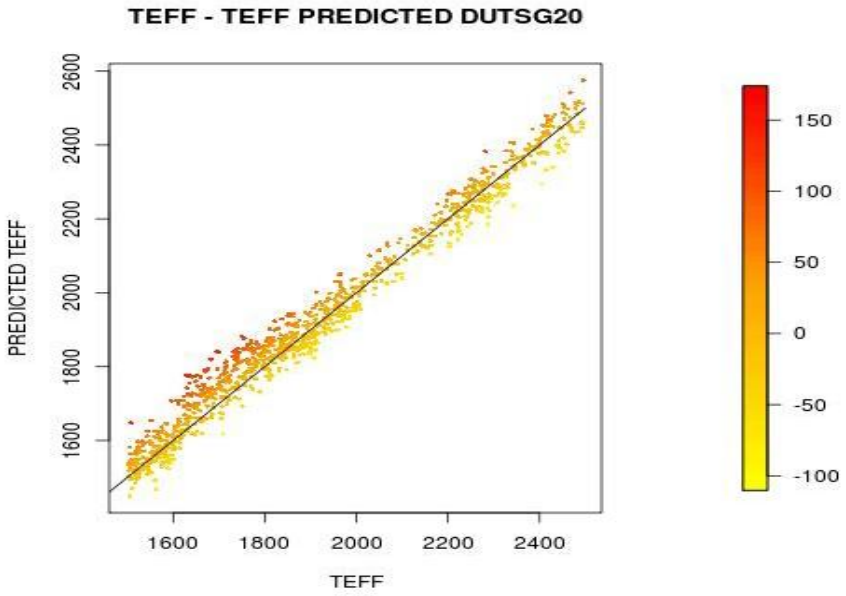


Figura. 71 Grafica de dispersión TeffvsTeffpredicted. sobreGPs para el conjunto de espectros DUSTG20area1.

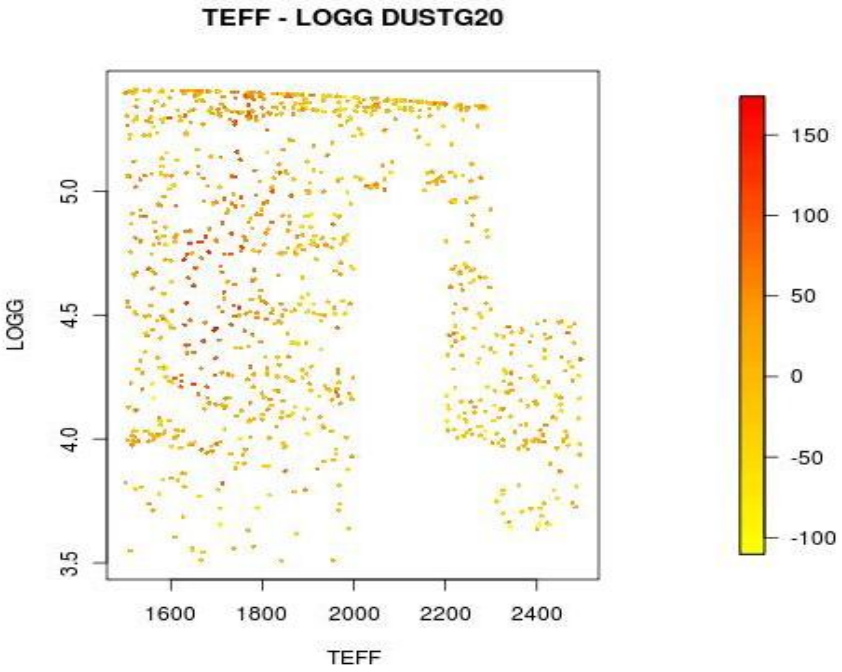


Figura. 72. Grafica de dispersión TeffvsLogg sobre GPs, para el conjunto de espectros DUSTG20area1

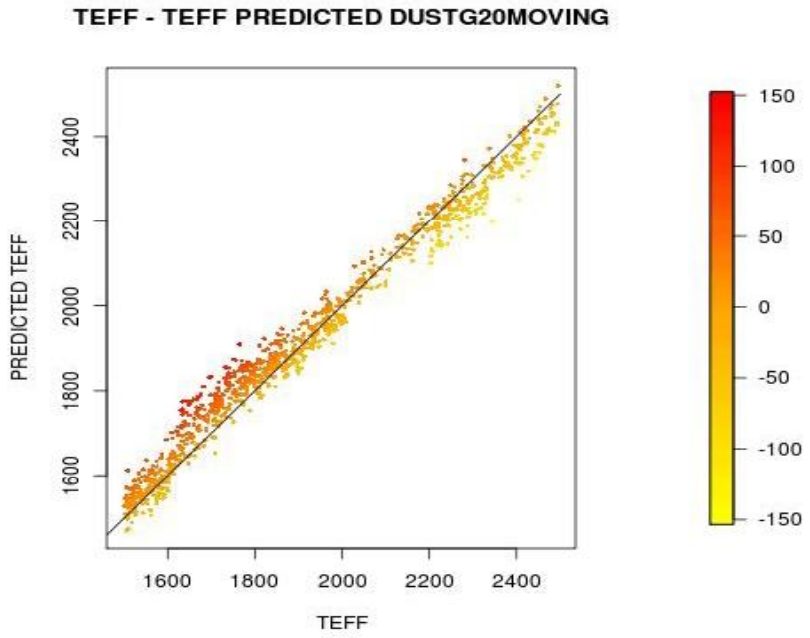


Figura 73 Gráfica de dispersión Teff vs Teff predicted sobre GPs para el conjunto DUSTG20areal moving.

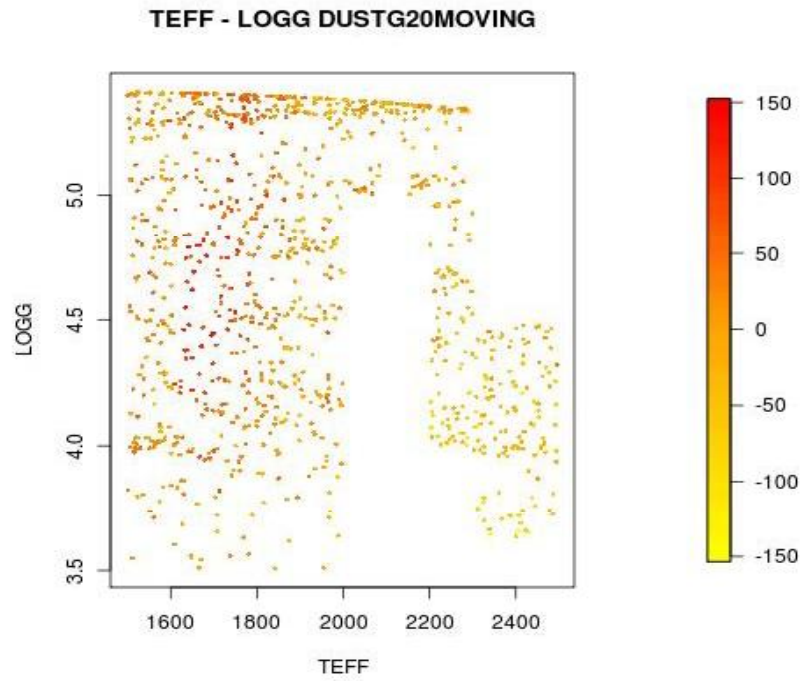


Figura 74. Gráfica de dispersión Teff vs Logg sobre GPs, para el conjunto de espectros DUSTG20areal moving

Por un lado, se observamos como para el conjunto de espectros DUSTG20area1 la aplicación del suavizado no supone un añadido, ya que el resultado es practicamente igual y las diferencias no se pueden apreciar con la información disponible.

Por los resultados anticipados sobre RANG15area1, y teniendo en cuenta que los modelos DUST son espectros con temperatura efectiva superior a 1500° K, las predicciones para temperatura real entre 1500° K hasta 1900° K son predicciones por encima de la temperatura real.

Por encima de los 1900° K, para los modelos DUST, las predicciones son mejores. Sin embargo, a partir de 2200 ° K, las predicciones son estimaciones por debajo de la temperatura real.

A continuación, se presenta un estudio del clasificador GPs sin reducción de dimensionalidad, para los conjuntos de validación CONDG202Y y DUSTG202Y, es decir, sobre conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación

La tabla 21, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG202Yarea1 y CONDG202Yarea1moving

| | CONDG202Y | CONDG202Y moving |
|---|---|---|
| Correlation coefficient | 0.9498 | 0.9662 |
| Mean absolute error | 97.5859 | 81.5273 |
| Root mean squared error | 128.4802 | 105.5823 |

Tabla 21: Resultados para GPs de los conjuntos de espectros CONDG202Yarea1

Las figuras 75 y 77 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG202Yarea1 y CONDG202Yarea1moving empleando el clasificadorGPs.

Las figuras 76 y 78 nos muestran respectivamente gráficas de dispersión de temperatura real frente

al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG202Yarea1 y CONDG202Yarea1moving empleando el clasificador de Procesos Gaussianos

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

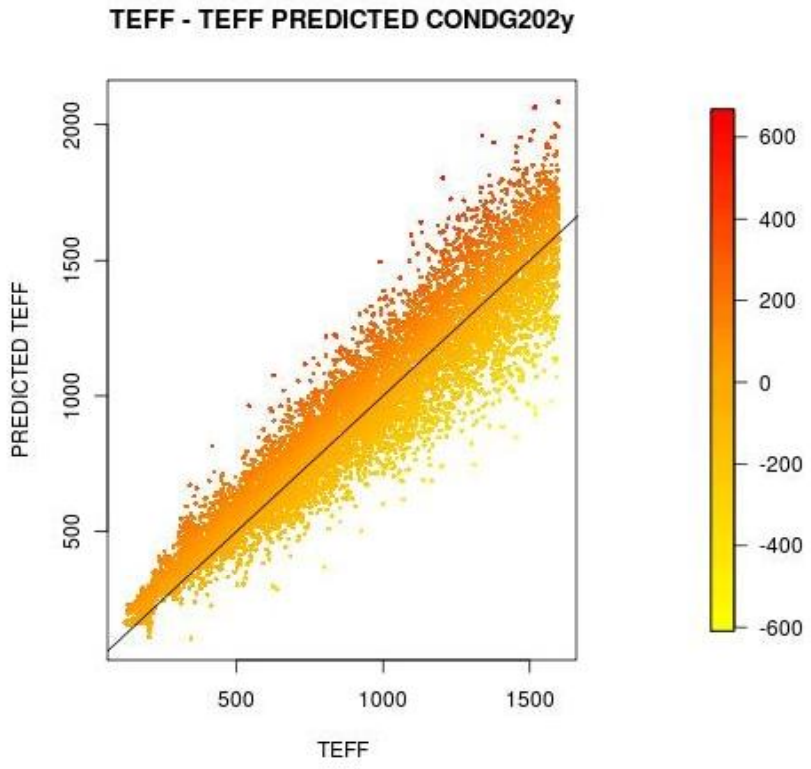


Figura 75 Gráfica de dispersión TeffTvs TeffTpredicted, sobre GPs para el conjunto de espectros CONDG202Yarea1.

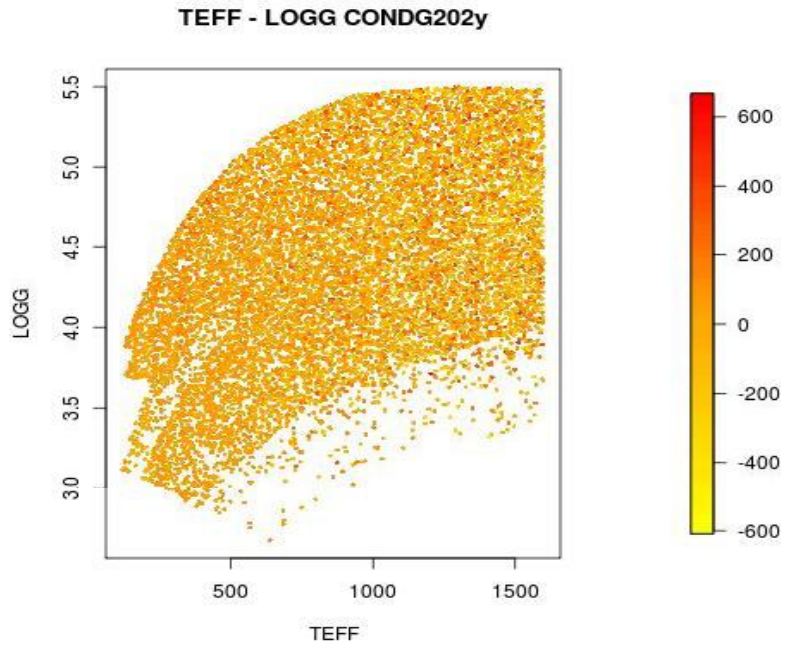


Figura 76. Grafica de dispersión Teff vs Logg sobre GPs, para el conjunto de espectros CONDG202Yareal

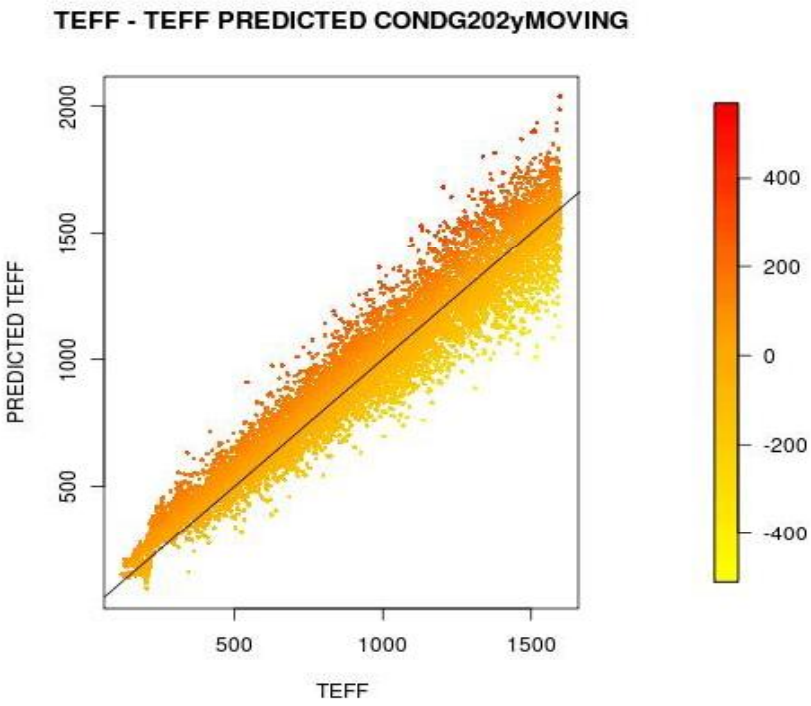


Figura 77 Grafica de dispersión Teff vs Teff predicted sobre GPs para el conjunto CONDG202Yareal moving.

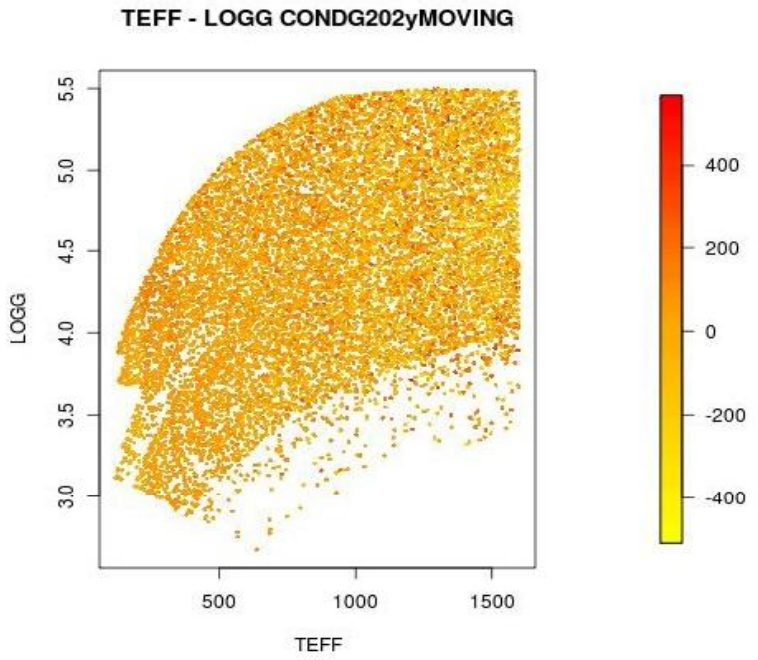


Figura 78. Gráfico de dispersión Teff vs Logg sobre GPs. para el conjunto de espectros CONDG20areaImoving

Se observa como para el conjunto de espectros CONDG202Yarea1, la aplicación del suavizado mejora los resultados, tanto en el error medio como en los valores máximos de error

Sin embargo, los errores máximos son tan grandes que no podemos hablar de que este clasificador pueda emplearse en fases intermedias de la misión

A continuación, la tabla 22 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG202Yarea1 y DUSTG202Yarea1moving

| | DUSTG202Y | DUSTG202Y moving |
|---|---|---|
| Correlation coefficient | 0.8197 | 0.8642 |
| Mean absolute error | 143.9654 | 118.0826 |
| Root mean squared error | 180.5085 | 147.8489 |

Tabla 22: Resultados para GPs de los conjuntos de espectros DUSTG202Yarea1, DUSTG202Yarea1moving

Las figuras 79 y 81 nos muestran respectivamente gráficas de dispersión de temperatura estimada

frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificadorGPs.

Las figuras 80 y 82 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificadorGPs

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

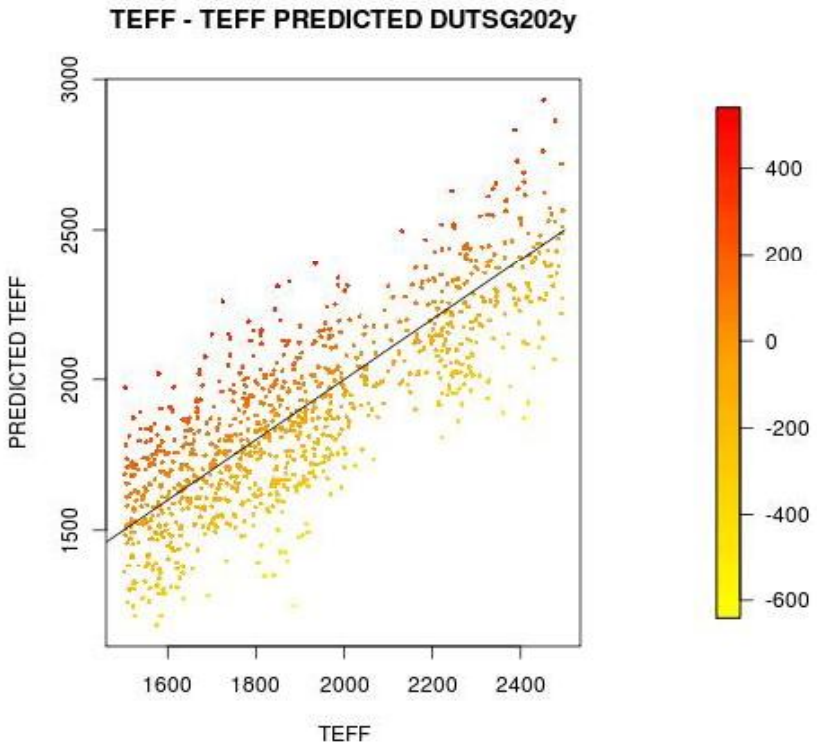


Figura 79 Gralica de dispersión Teff vs Teff predicted, sobre GPs para el conjun.o de espectros DUSTG202Yarea1.

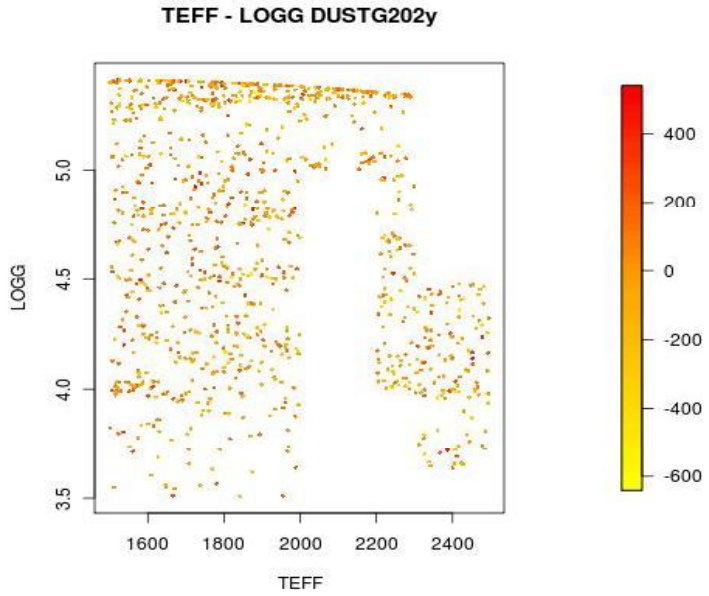


Figura 80. Gráfica de dispersión TeffVs Logg sobre KNN, para el conjunto de espectros DUSTG202Yareal

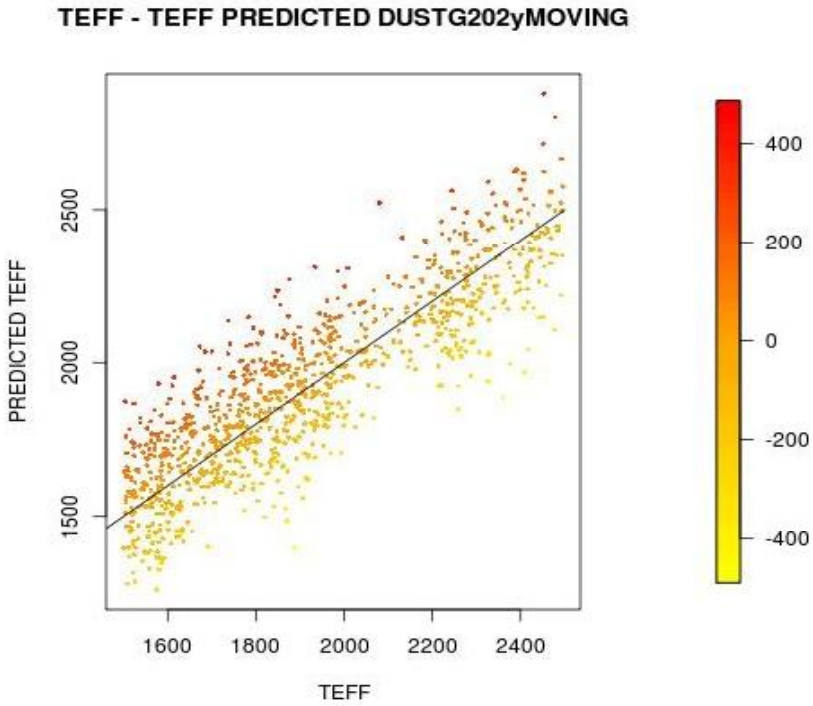


Figura 81 Gráfica de dispersión Teff vs Teff predicted, sobre GPs para el conjunto DUSTG20areal moving.

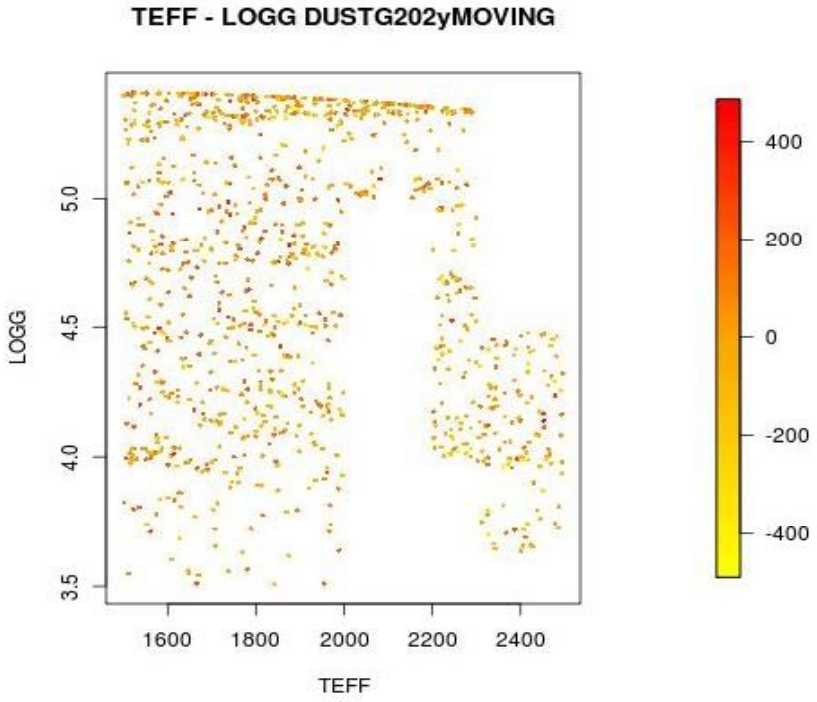


Figura 82. Gráfica de dispersión Teff vs Logg sobre GPs, para el conjunto de espectros DUSTG20area1moving

Los conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación, por lo tanto, es normal que los resultados obtenidos sean peores que los conjuntos de datos de validación DUSTG20area1

Para el conjunto de espectros DUSTG202Yarea1, al igual que para el conjunto de espectros estudiados de CONDG202Yarea1 para GPs, la aplicación del suavizado mejora los resultados en el error cuadrático medio, sin embargo, al igual que pasa con CONDG202Yarea1, los errores máximos son tan grandes que hacen inviable considerar este clasificador para periodos intermedios de la misión.

#### 3.2.1.4 Conclusiones Parciales

No se puede evaluar y decidir sobre cuál clasificador se comporta mejor hasta que no se realize un estudio más detallado, por ejemplo infiriendo los parámetros mediante técnicas bayesianas o aplicando el T-Student sobre determinados clasificadores.

Este tipo de estudio más detallado no es objeto del Trabajo fin de máster sino de su investigación futura que se desarrollará mediante la ejecución de una tesis doctoral.

Los resultados nos muestran que para magnitud aparente G20 y sin reducción de dimensionalidad, los tres clasificadores tienen un comportamiento parecido, si bien para temperaturas entre 100º y 1600º K (modelos COND) son las máquinas de vectores soporte las que a priori parecen aportar mejores soluciones, los Procesos Gausianos tienen unos máximos de error más pequeños, mientras que los k vecinos mas cercanos se comportan de una forma más regular.

Hay que notar que para los clasificadores SMO y KNN (para GP no tanto) que por debajo de 500 grados Kelvin los resultados mostrados hacen sugerir un fallo en la interpolación de los datos proporcionados por DPAC-CU2. Este detalle particular conviene ser revisado correspondientemente con el grupo de investigación

Para las temperaturas entre 1500 y 2500 K (modelos DUST) los K-NN presenta mejores predicciones, pendiente de realizar su correspondiente validación mediante inferencias bayesianas o aplicando T-Student.

Para etapas intermedias de la misión en las que el número de observaciones de una estrella dada sea inferior al total, en las cuales la relación señal/ruido será muy inferior (conjunto de datos simulados con etiqueta G202Y), el sistema predictivo k-NN con moving average sobre los datos aparentemente obtiene mejor resultado indiferentemente de la teff esperada (tanto para los modelos COND como DUST)

Tanto para SMO como para GPS los errores proporcionados son muy elevados y no pueden emplearse los clasificadores para obtener información de predicción.

### 3.2.2 Clasificadores con transformación y reducción de atributos

A continuación, se pretende evaluar si se puede conseguir mejorar los resultados obtenidos por los clasificadores anteriores, aplicando reducción de dimensionalidad sobre los atributos de los espectros

Para ello, se emplearán diferentes métodos (definidos matemáticamente en los apartados 2.7 y 2.9 de la presente memoria) empleando las transformadas PCA (experimentos descritos en el apartado 3.2.2.1), DiffusionMaps (experimentos descritos en el apartado 3.2.2.2) y KPLS (experimentos descritos en el apartado 3.2.2.3)

Hay que considerar, que en el caso de KPLS la transformada incluye también el método de clasificación.

Por lo tanto los clasificadores a evaluar en este apartado son:

En el apartado 3.2.2.1, los KNN, SMO y GPs sobre la transformada PCA sobre Weka

En el apartado 3.2.2.2, los KNN, SMO y Gps sobre la transformada DiffusionMaps en R y clasificador en Weka.

En el apartado 3.2.2.3, el transformador/clasificador KPLS sobre R

Para entrenamiento de los clasificadores se emplea el conjunto de datos NOMarea1 (conjunto de espectros NOM normalizados a area1).

El conjunto de espectros referido como RANG15 se trata del conjunto de espectros proporcionados por France Allard, filtrados por DPAC-CU2, sin discriminación entre los modelos COND y DUST.

Los conjuntos de espectros referidos a los modelos CONDarea1 y DUSTarea1 para magnitud aparente G20 se referencian como CONDG20, DUSTG20

Los conjuntos de espectros que muestran los primeros resultados de la sonda GAIA a mitad de observación se referencian como CONDG202Y y DUSTG202Y

Para validación de los clasificadores se emplean los conjuntos de espectros RANG15, CONDG20, DUSTG20, CONDG202Y y DUSTG02Y aplicando sobre ellos la normalización a area1, obteniendo los conjuntos detallados en la tabla 7.

También en la tabla 7, se detallan otros conjuntos de espectros empleados para validar los clasificadores. Estos conjuntos de espectros, etiquetados con la coletilla "movingav", se han obtenido partiendo de los anteriores conjuntos de espectros CONDG20area1, DUSTG20area1, CONDG202Yarea1 y DUSTG202Yarea1 (espectros con ruido) aplicando sobre ellos las técnicas de Moving Average.

En los clasificadores que ofrecen la representación por medio de funciones Kernel, se realizó una experimentación inicial empleando NOMarea1 con validación cruzada.

El resultado de esta experimentación, determinó que tanto para las máquinas vectores soporte como para procesos gausianos se comportan mejor usando un kernel basado en funciones de Base Radial Gaussiana (RBF) para obtener el nuevo espacio transformado .

La experimentación para la predicción para PCA se realizó mediante el software Weka. Para los Mapas de Difusión se empleó R para la reducción de dimensionalidad y Weka para la ejecución mediante clasificadores. Para KPLS tanto la reducción como la clasificación se realizó empleando el paquete estadístico R

A continuación se detallan las variables que se optimizaron para cada clasificador

Para los K-vecinos cercanos::

- KNN Número de vecinos cercanos a usar.

  DistanceWeighting Método de ponderación de la distancia entre vecinos.

  NearestNeighbourSearchAlgorithm. Algoritmo de búsqueda del vecino más cercano.

Para las máquina de vectores soporte (SMO) fueron

Margen blando (variable C), al no existir una separación perfecta entre los hiperplanos, el parámetro C controla la compensación entre errores de entrenamiento y los márgenes rígidos, creando así un margen blando (soft margin) que permite algunos errores en la clasificación a la vez que los penaliza.

- Factor Gamma (variable g) del nucleo kernel RBF.

Para los clasificadores basados en Procesos Gausianos

- Noise Nos determina el nivel del Ruido Gausiano (el cual es añadido la diagonal de la Matriz de Covarianza)

- Factor Gamma (variable g) del nucleo kernel RBF

Para la optimización de todos los clasificadores y sus variables, se emplearon los conjuntos de espectros NOMarcal para entrenar, y RANG15marcal para validar

La búsqueda de los valores óptimos de los parámetros de cada clasificador se realizó empleando un conjunto de acciones repetitivas:

1) Asignando a las variables unos valores iniciales igual a 1.

2) Entrenamiento y validación del sistema.

3) Observación de los resultados obtenidos.

4) Comparativa de variables con mejor resultado obtenido hasta el momento.

5) Escalado / modificando los valores de las variables en función de 4)

6) Retorno al punto 2)

Este bucle iterativo se repitió hasta localizar las variables del clasificador que definían un comportamiento predictivo óptimo frente a la clase Temperatura efectiva (teff) de los datos de validación

Para las aproximaciones iniciales en validación cruzada con NOMareal, se obtuvieron diferentes valores de las variables de los clasificadores que las obtenidas para los modelos predictivos entrenados con los conjuntos de entrenamiento NOMareal y validados por lo conjuntos de validación RANG15areal.

Como ya se ha comentado, para la optimización de todos los clasificadores y sus variables, se emplearon los conjuntos de espectros NOMareal para entrenar, y RANG15areal para validar.

La reevaluación de los modelos y obtención de resultados del clasificador, consistió en introducir los conjuntos de espectros con ruido (CONDG20, CONDG20movingav, DUSTG20, DUSTG20movingav, CONDG202Y, CONDG202Ymovingav, DUSTG202Y, DUSTG202Ymovingav) a la entrada de los modelos clasificadores obtenidos, sin reajuste del clasificador, y por lo tanto, manteniendo el valor de las variables obtenidas en el clasificador en las condiciones de entrenamiento con NOMareal y validación con RANG15areal

#### 3.2.2.1 Resultados para Clasificadores con Preprocesado PCA

Como se ha comentado en el apartado *2.7.1 Análisis de Componentes Principales*, las componentes principales son combinaciones lineales de las variables originales tales que cada componente retenga el máximo de información (varianza) de las variables originales, sin que estos nuevos nuevos componentes estén correlacionados entre sí

Un aspecto clave en PCA es la interpretación de los atributos ya que ésta no viene dada a priori, sino que será deducida tras observar la relación de los factores con las variables iniciales (habrá pues, que estudiar tanto el signo como la magnitud de las correlaciones).

Esto no siempre es fácil, y será de vital importancia el conocimiento que el experto tenga sobre la materia de investigación.

Tal y como se observa en la Tabla 11. En primer lugar nos aparecen los valores propios (eigenvalue) de cada componente principal, proporción de varianza explicada (proportion) por cada una de ellos y la varianza explicada acumulada (cumulative)

Los datos de varianza explicada son muy importantes para saber cuántos componentes principales vamos a utilizar en nuestro análisis. No hay una regla definida sobre el número que se debe utilizar, con lo cual deberemos decidir en función del número de variables iniciales (hay que recordar que se trata de reducirlas en la medida de lo posible) y de la proporción de varianza explicada acumulada

Al aplicar PCA sobre los conjuntos de datos, se determina seleccionar el número de atributos que cubran al menos el 95% de la varianza.

De esta forma, aplicando PCA en weka mediante el comando

*weka.filters.unsupervised.attribute.PrincipalComponents*

reduce la dimensionalidad de 180 a 6 componentes

| COMPONENT'S | EIGENVALUE | PROPORTION | CUMULATIV'S |
|---|---|---|---|
| V1 | 85.63215 | 0.5403 | 0.5403 |
| V2 | 30.26525 | 0.17099 | 0.71129 |
| V3 | 23.54097 | 133 | 0.84429 |
| V4 | 6.09575 | 0.03952 | 0.88381 |
| V5 | 6.13889 | 0.03468 | 0.91849 |
| V6 | 5.62917 | 0.0318 | 0.95503 |

Tabla 23. EigenAnálisis de la Matriz Correlación

Finalmente, aparecen las correlaciones de cada componente principal con cada variable: esto nos ayudará a interpretar las variables.

```
Eigenvectors
 V1       V2       V3       V4       V5       V6
-0.09    -0.0395   0.0593  -0.0094  -0.0425   0.0687 l1
-0.0868  -0.0346   0.0707   0.0032  -0.0845   0.0668 l2
-0.085   -0.0203   0.0814  -0.0109  -0.1149   0.071  l3
-0.0856  -0.012    0.0837  -0.0322  -0.1137   0.0751 l4
-0.0878  -0.0205   0.0779  -0.0357  -0.0823   0.0709 l5
-0.089   -0.0413   0.0707  -0.0163  -0.0399   0.0575 l6
-0.0877  -0.06     0.0689   0.0123  -0.0059   0.0423 l7
-0.0851  -0.0647   0.0765   0.0359   0.0114   0.0333 l8
-0.083   -0.0481   0.0939   0.048    0.0109   0.0332 l9
-0.081   -0.0089   0.1131   0.0427  -0.008    0.0362 l10
-0.0797   0.0294   0.116    0.0217  -0.0342   0.0309 l11
-0.0836   0.0396   0.101    0.0028  -0.0485   0.0172 l12
-0.0908   0.0289   0.0784  -0.0035  -0.0464   0.0049 l13
-0.0966   0.0165   0.0548   0.0018  -0.0343  -0.0003 l14
-0.0995   0.0138   0.0341   0.0137  -0.0202   0.001  l15
-0.0996   0.0232   0.0206   0.0252  -0.0108   0.0044 l16
-0.0978   0.0383   0.019    0.0269  -0.0128   0.0039 l17
-0.0963   0.0467   0.0262   0.0158  -0.0256  -0.0019 l18
-0.0964   0.0444   0.0285   0.0025  -0.0388  -0.0062 l19
-0.0978   0.0395   0.0177  -0.0024  -0.0443  -0.0024 l20
-0.0984   0.0389  -0.0038   0.001   -0.0395   0.0114 l21
```

Figura 83: Muestra parcial de la tabla de correlación componentes PCA / atributos originales

La componente principal V1, por ejemplo está correlada negativamente con las primeras 120 variables y correlada positivamente con las 60 últimas.

También se obtiene el gráfico en dos dimensiones de PC1 y PC2 (figura 84), PC1 y PC3 (figura 85), PC1 y PC4 (figura 86), PC1 y PC5 (figura 87), PC1 y PC6 (figura 88) donde podemos ver la variabilidad de las observaciones, y si existe alguna que ofrezca un valor extrañamente alto o bajo en cada eje.

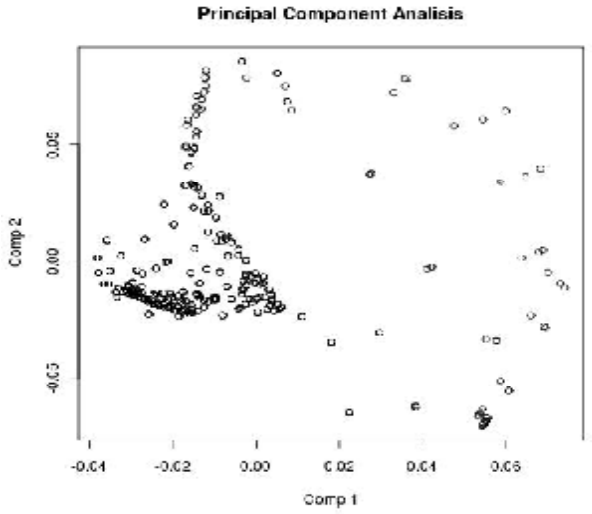


Figura 84 : Análisis de Componentes Principales sobre NOM, gráfica Componente 1 Componente 2

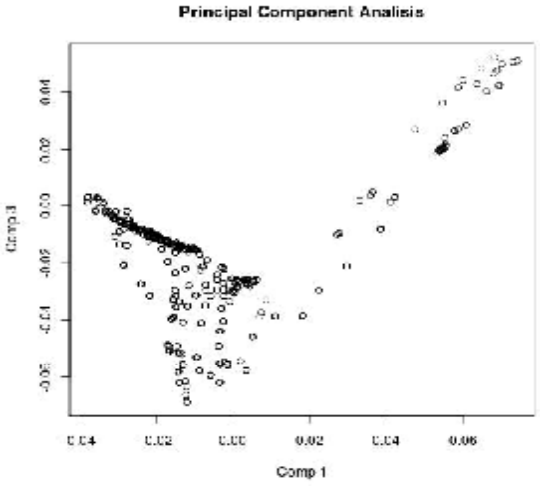


Figura 85 : Análisis de Componentes Principales sobre NOM, gráfica Componente 1 Componente 3

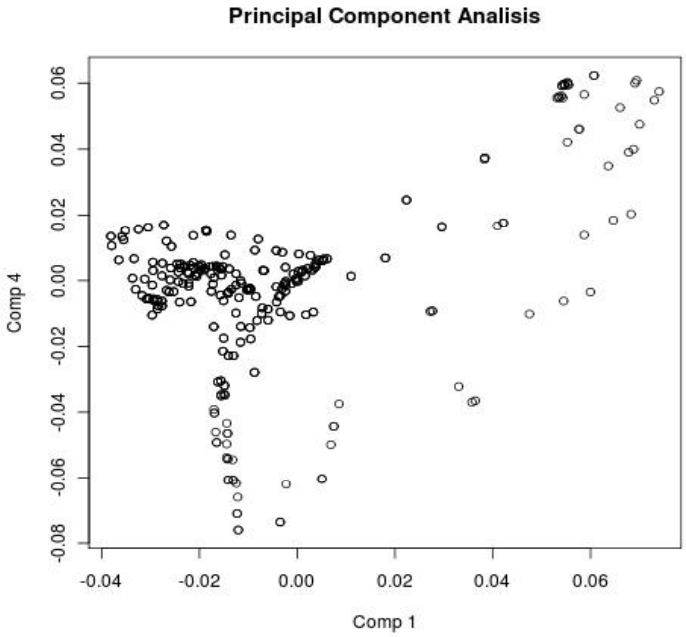


Figura 86 : Analisis de Componentes Principales sobre NOM, gráfica Componente 1 Componente 4

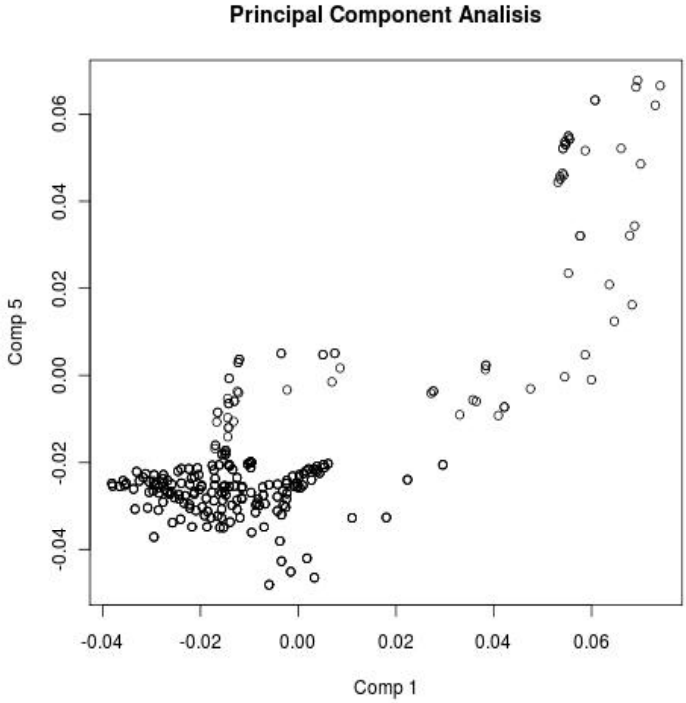


Figura 87 : Analisis de Componentes Principales sobre NOM, gráfica Componente 1 Componente 5

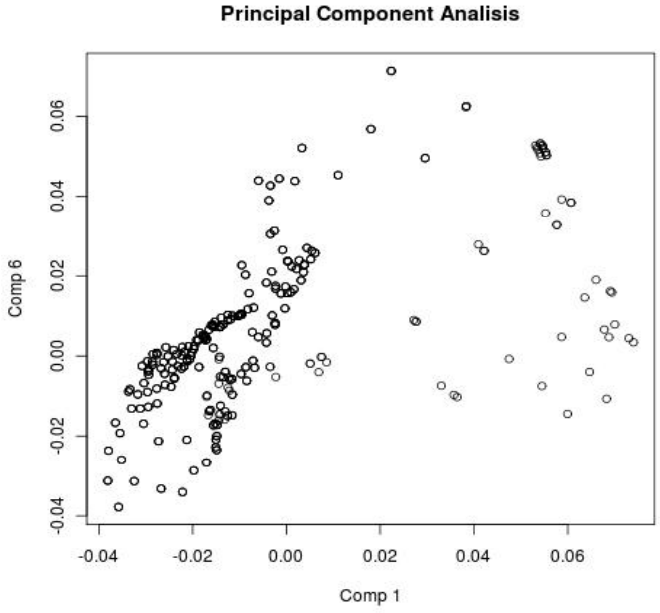


Figura 88 : Analisis de Componentes Principales sobre NOM, gráfica Componente 1 Componente 6

Al aplicar PCA en Weka, hay que tener en cuenta no normalizar los datos porque ya se han normalizado a area1 previamente

#### 3.2.2.1.1 Resultados para el Clasificador K-NN

Se ha optimizado la aplicación de K vecinos cercanos sobre los datos en Weka, empleando el siguiente comando

*weka.classifiers.meta.AttributeSelected1.classifier*

Definiendo como Clasificador "ibk", con las siguientes variables:

- KNN Número de vecinos cercanos a usar: 4

  DistanceWeighting, Método de ponderación de la distancia entre vecinos: la inversa de la distancia *"Weight by 1/distance"*

NearestNeighbourSearchAlgorithm. Algoritmo de búsqueda del vecino más cercano: CoverTree.

La Tabla 23 muestra los resultados obtenidos tanto para validación cruzada como para la validación con RANG15area1

| | NOMarea1 | RANG15 |
|---|---|---|
| Correlation coefficient | 0.9999 | 0.9863 |
| Mean absolute error | 0.7092 | 43.701 |
| Root mean squared error | 8.4215 | 82.5992 |
| Relative absolute error | 0.1229 % | 9.6608 % |
| Root relative squared error | 1.2402 % | 15.1546 % |

Tabla 23: Resultados sobre PCA_KNN para validación cruzada y para el conjunto de validación sin ruido RANG15area1.

Nos apoyamos en las gráficas de dispersión sobre la validación con RANG15area1 (figuras 88 y 89) para extraer conclusiones. Téngase en cuenta que el degradado de color muestra el error en la predicción de la temperatura con respecto a la real

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

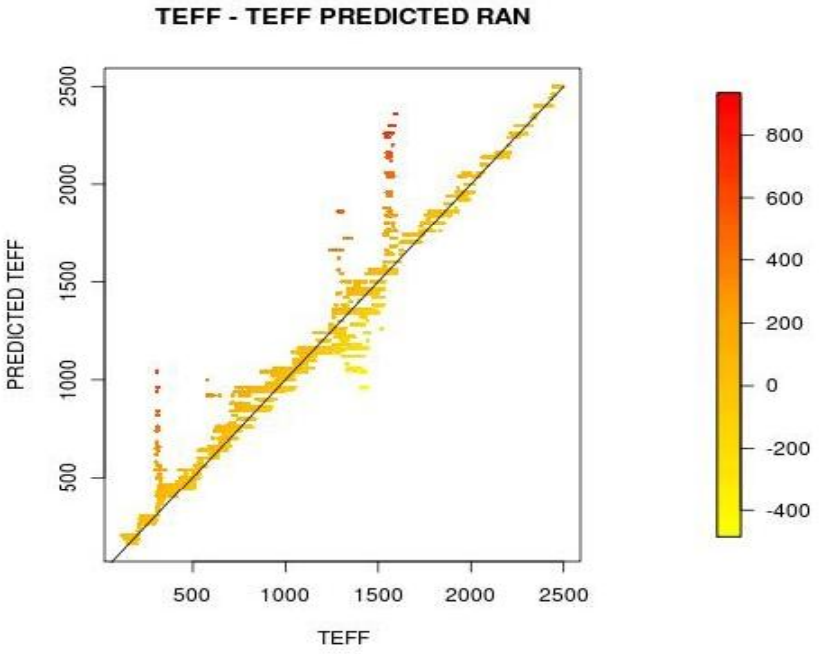


Figura 88 Gratica de dispersión para la predicción sobre PCA KNN de TEFF vs TEFF predicted para el conjunto de validacion RANarea1

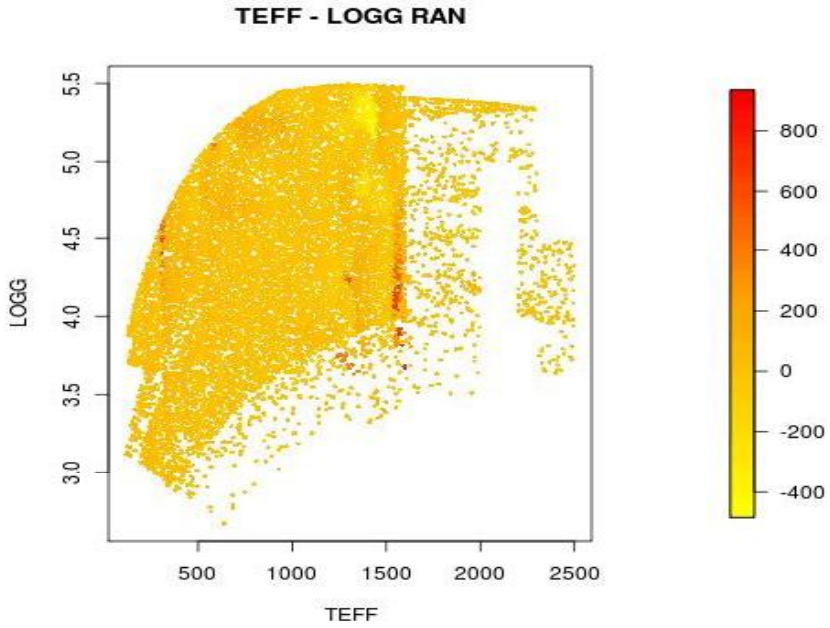


Figura 89 Gratica de dispersión para la predicción sobre PCA−KNN de TEFF vs LOG para el conjunto de validación RANarea1.

Sobre la figura 88, se observa el ya comentado efecto de escalonado en los valores de temperatura por debajo de 500 grados Kelvin. Este escalonado como se ha comentado anteriormente, puede coincidir con un error de interpolación en los modelos RAN generados por DPAC-CU2, sobre los modelos NOM (cuyos valores nominales de temperatura efectiva van de 100 en 100)

Para este clasificador, se observa tanto en la figura 88 como 89 que, las mejores predicciones se encuentran para el rango de temperatura por encima de 1600º K.

Un efecto de aplicar las Componentes Principales, tal y como se comenta en el apartado 2.7.1 y en la figura 8, es que la información más geométrica se pierde al proyectar los datos sobre el eigenvalor de mayor variabilidad. Este efecto se hace visible alrededor de las temperaturas reales 400º, 1400º y 1600º K, existe un error de predicción por encima de los 800 º K. Parece, y posteriormente se comprobará que los errores vienen de la trasnformación PCA del conjunto de espectros para validación COND

Este tipo de error provoca que el clasificador no sea capaz de estimar correctamente ninguna temperatura, ya que una temperatura efectiva estimada por el clasificador, por ejemplo, de 2400º Kelvin puede estar haciendo referencia a una temperatura real de 1600º K.

Sin embargo, si pudiera demostrarse que el error de predicción es sobre la aplicación de PCA sobre los modelos COND,por encima de los 1600º K, es decir, para predicciones de modelos DUST, el clasificador muestra unos resultados muy fiables.

Para que este clasificador fuera operativo, deberiamos de poder filtrar previamente los conjuntos de espectros para validación con temperaturas reales por encima de los 1600º Kelvin.

A continuación, en un estudio más profundo se va a reevaluar el clasificador para los conjuntos de espectros con ruido comentados en la tabla 7

La tabla 24, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG20area1 y CONDG20area1moving

| | CONDG20 | CONDG20 moving |
|---|---|---|
| Correlation coefficient | 0.9769 | 0.9736 |
| Mean absolute error | 48.0285 | 52.5454 |
| Root mean squared error | 87.3842 | 93.8512 |
| Relative absolute error | 11.1819 % | 12.2335 % |
| Root relative squared error | 16.7321 % | 17.9704 % |

Tabla 24: Resultados para PCA+KNN de los conjuntos de espectros CONDG20area1 y CONDG20area1moving

Las figuras 90 y 92 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG20area1 y CONDG20area1moving empleando el clasificador PCA+KNN

Las figuras 91 y 93 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG20area1 y CONDG20area1moving empleando el clasificador PCA + KNN

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

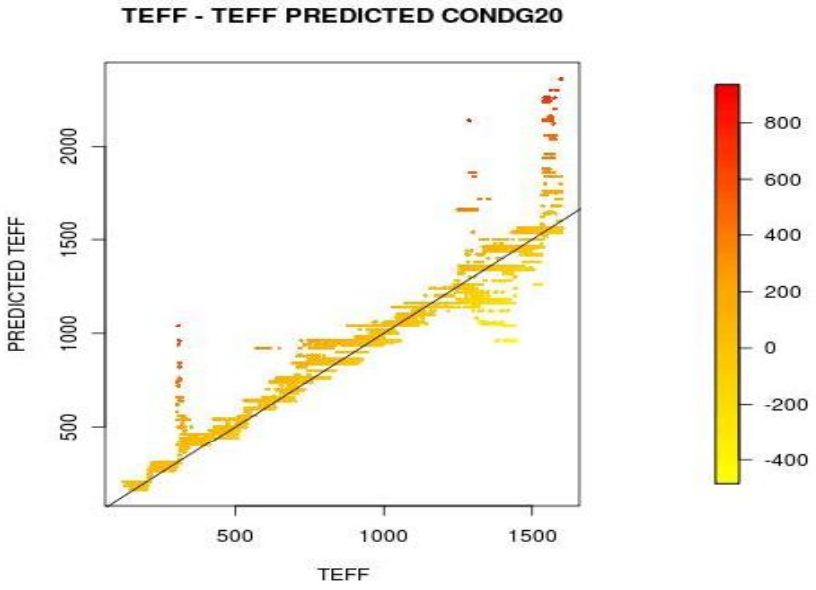


Figura 90

Gralica de dispersión TeIT vs TeIT predicted, sobre PCA_KNN para el conjunto de espectros CONDG20area1.

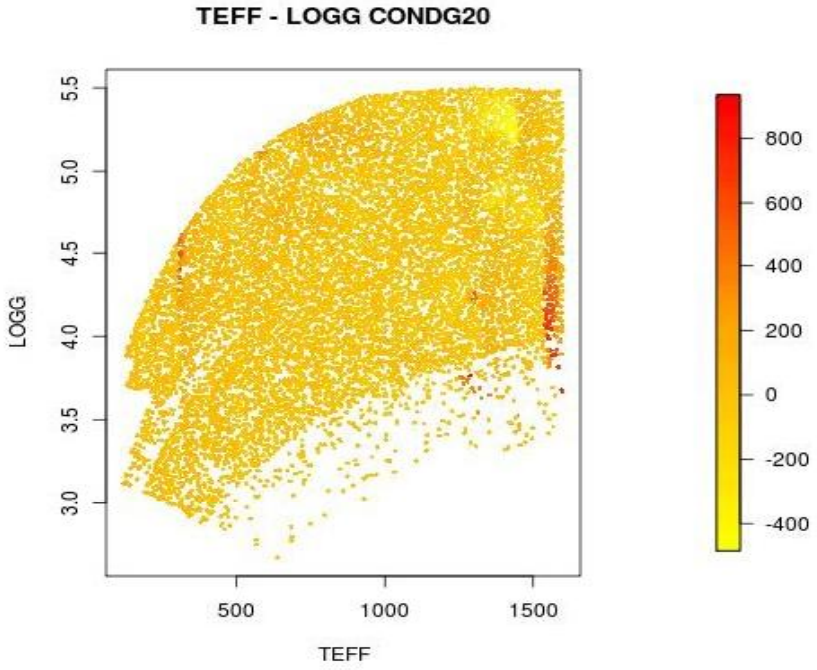


Figura 91. Gralica de dispersión TeIT vs Logg sobre PCA_KNN, para el conjunto de espectros CONDG20area1

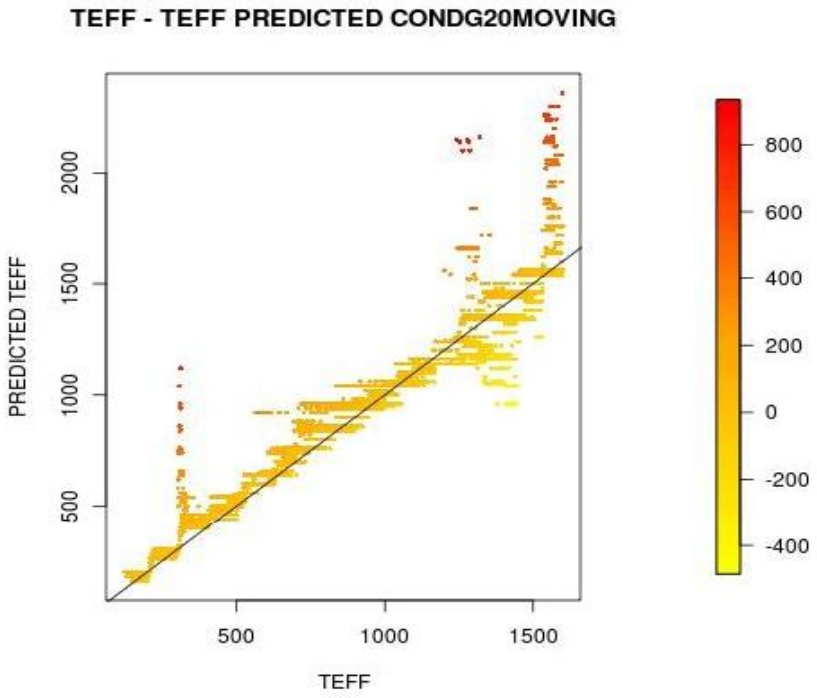


Figura 92 Gráfica de dispersión Teff vs Teff predicted, sobre PCA+KNN para el conjunto CONDG20area1moving.

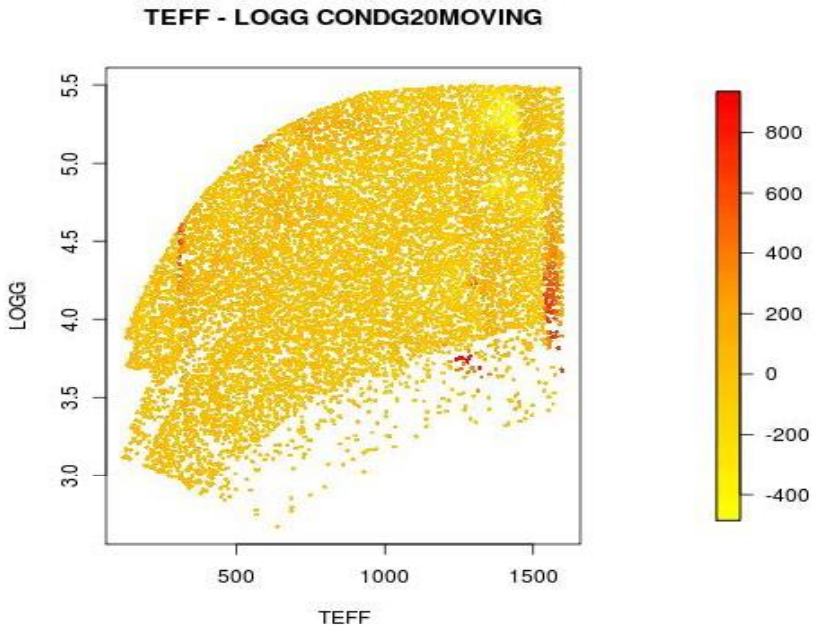


Figura 93. Gráfica de dispersión Teff vs Logg sobre PCA+KNN, para el conjunto de espectros CONDG20area1moving

Por un lado, observamos como, para el conjunto de espectros CONDG20area1, la aplicación del suavizado no mejora los resultados.

Ocurren los mismos problemas detectados para los conjuntos de datos de validación sin ruido RANG15area1 para el rango de datos hasta 1500 ° K

- El escalonado en temperaturas inferiores a 500 ° K.

- En las temperaturas reales 400°, 1400° y 1600° K, existe un error de predicción por encima de la temperatura real de los 800 °.

Además, se observa como, en la temperatura real de 1300° K, existen errores de predicción por debajo de la temperatura real con valores de - 400 ° K.

Como se ha visualizado en las gráficas 88 y 89, estos errores están muy focalizados en

La temperatura real de 400° Kelvin con un logaritmo de la gravedad de aprox. 4,5

- La temperatura real de 1400° Kelvin, con un logaritmo de la gravedad de aprox 3,5
- La temperatura real de 1600° Kelvin, con un logaritmo de la gravedad en valores entre 3,5 y 4,5.
- Alrededor de la temperatura real 1300° Kelvin, para logaritmos de la gravedad entre 4,5 y 5,5.

Tal y como se ha comentado anteriormente, es posible que el error venga debido a la perdida de información al aplicar la transformada PCA. Deberia plantearse la aplicación de un mayor número de componentes en la reducción de dimensionalidad

De la misma forma, para que este clasificador fuera operativo, deberíamos de poder filtrar inicialmente conjuntos de datos con temperaturas reales por encima de los 1600 Kelvin (modelos DUST), eliminando cualquier relación con los modelos COND

A continuación, la tabla 25 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG20area1 y DUSTG20area1moving:

| | DUSTG20 | DUSTG20 moving |
|---|---|---|
| Correlation coefficient | 0.9935 | 0.993 |
| Mean absolute error | 27.2111 | 26.9611 |
| Root mean squared error | 34.1996 | 33.7732 |
| Relative absolute error | 3.9977 % | 3.961 % |
| Root relative squared error | 4.6528 % | 4.5948 % |

Tabla 25: Resultados para PCA+KNN de los conjuntos de espectros DUSTG20area1 y DUSTG20area1moving

Las figuras 94 y 96 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos DUSTG20area1 y DUSTG20area1moving empleando el clasificador PCA + KNN.

Las figuras 95 y 97 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos para validación DUSTG20area1 y DUSTG20area1moving empleando el clasificador PCA+KNN

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

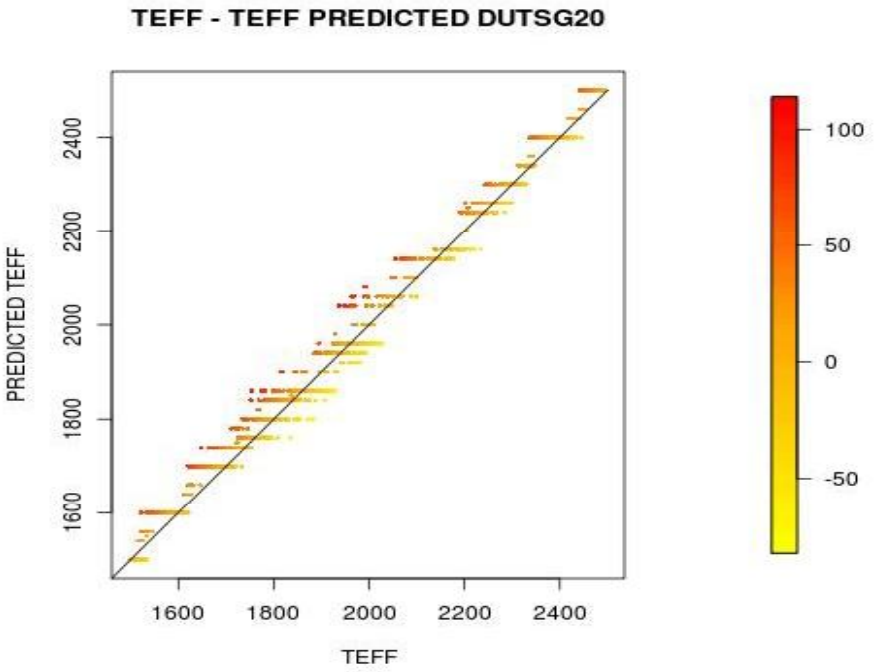


Figura 94 Grafica de dispersión Teff vs Teff predicted, sobre PCA KNN para el conjunto de espectros DUSTG20area1.

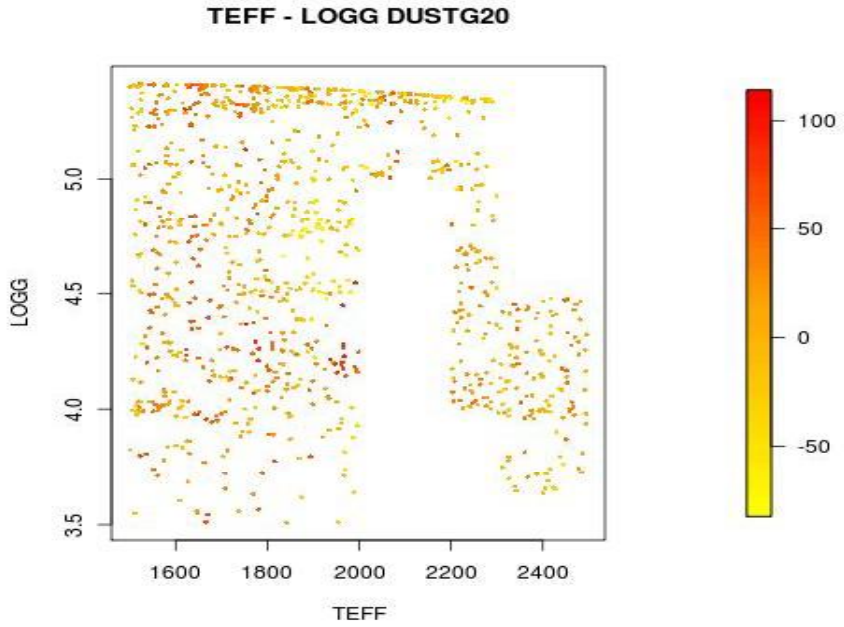


Figura 95. Grafica de dispersión Teff vs Logg sobre PCA–KNN, para el conjunto de espectros DUSTG20area1

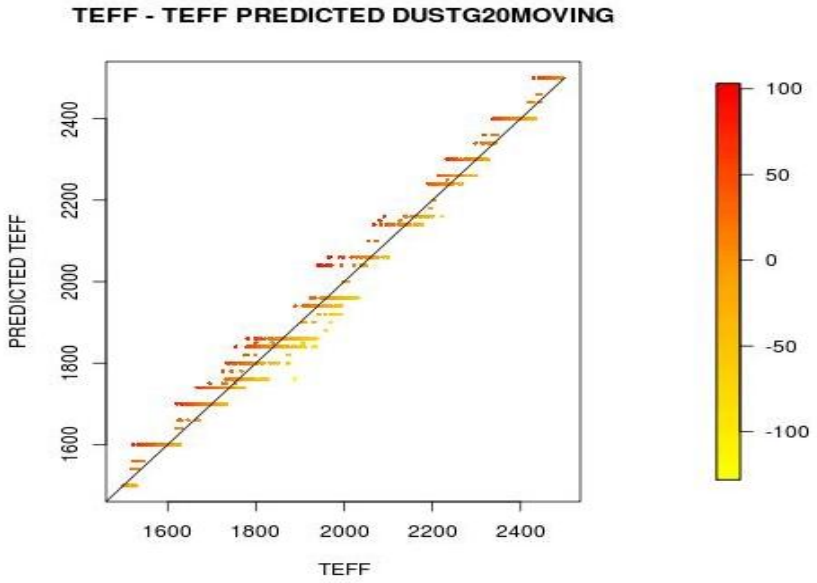


Figura 96 Gráfica de dispersión TeT vs TeH predicted, sobre PCA KNN para el conjunto DUSTG20area1moving.

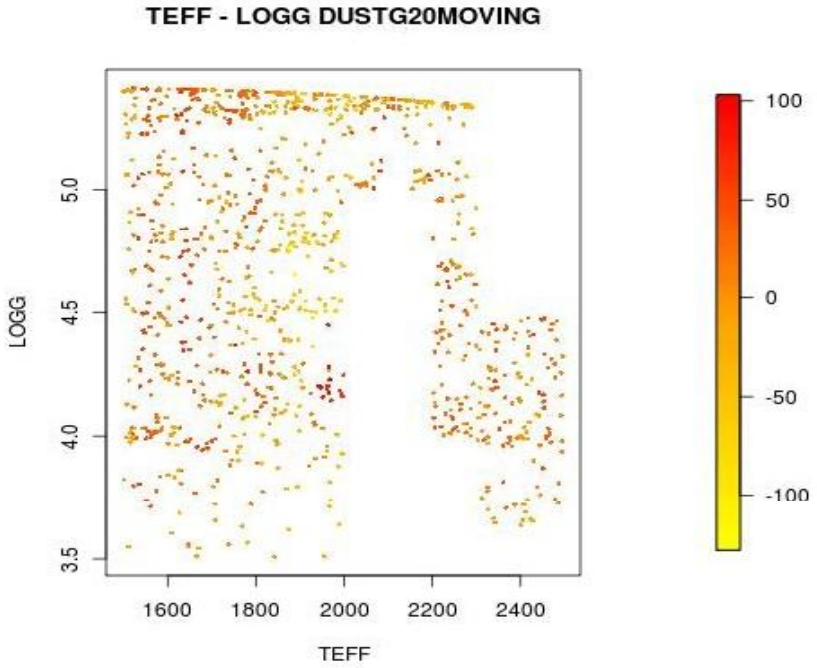


Figura 97. Gráfica de dispersión TeT vs Logg sobre PCA−KNN, para el conjunto de espectros DUSTG20area1moving

Por un lado, se observamos como para el conjunto de espectros DUSTG20area1 la aplicación del suavizado no supone un añadido, al contrario, aparecen errores más grandes de predicción por debajo de la temperatura real

Por los resultados anticipados sobre RANG15area1, y teniendo en cuenta que los modelos DUST son espectros con temperatura efectiva superior a 1500° Kelvin, las predicciones para temperatura real superior a 1600° Kelvin son las que mejor realiza este clasificador.

Se demuestra pues, que el error es debido a los modelos COND. Lo que queda pendiente de demostrar es si es debido a una perdida de información en la transformación PCA

Como se ha comentado antes, si pudieramos filtrar los modelos COND previamente, este clasificador podría ser valido, únicamente para modelos DUST.

A continuación, se presenta un estudio del clasificador PCA+KNN, para los conjuntos de validación CONDG202Y y DUSTG202Y, es decir, sobre conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación

La tabla 26, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG202Yarea1 y CONDG202Yarea1moving

| | CONDG202Y | CONDG202Y moving |
|---|---|---|
| Correlation coefficient | 0.969 | 0.9658 |
| Mean absolute error | 56.6718 | 60.4929 |
| Root mean squared error | 100.1663 | 105.8352 |
| Relative absolute error | 13.1942 % | 14.0838 % |
| Root relative squared error | 19.1796 % | 20.2651 % |

Tabla 26: Resultados para PCA+KNN de los conjuntos de espectros CONDG202Yarea1

Las figuras 98 y 100 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos

CONDG202Yarcal y CONDG202Yarcalmoving empleando el clasificador PCA+KNN.

Las figuras 99 y 101 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG202Yarcal y CONDG202Yarcalmoving empleando el clasificador PCA+KNN

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

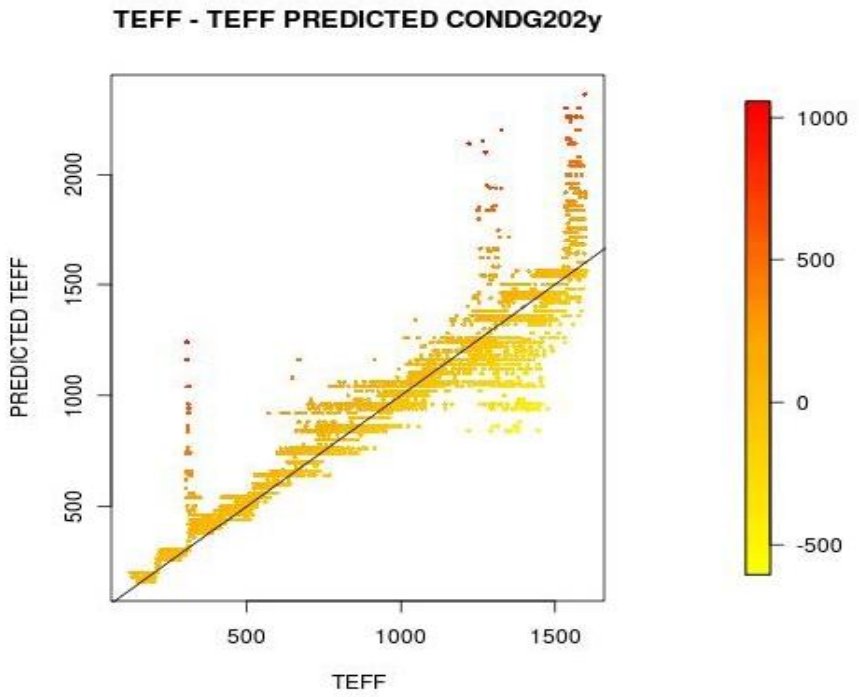


Figura 98 Gráfica de dispersión Teff vs Teff predicted, sobre PCA KNN para el conjunto de espectros CONDG202Yarcal.

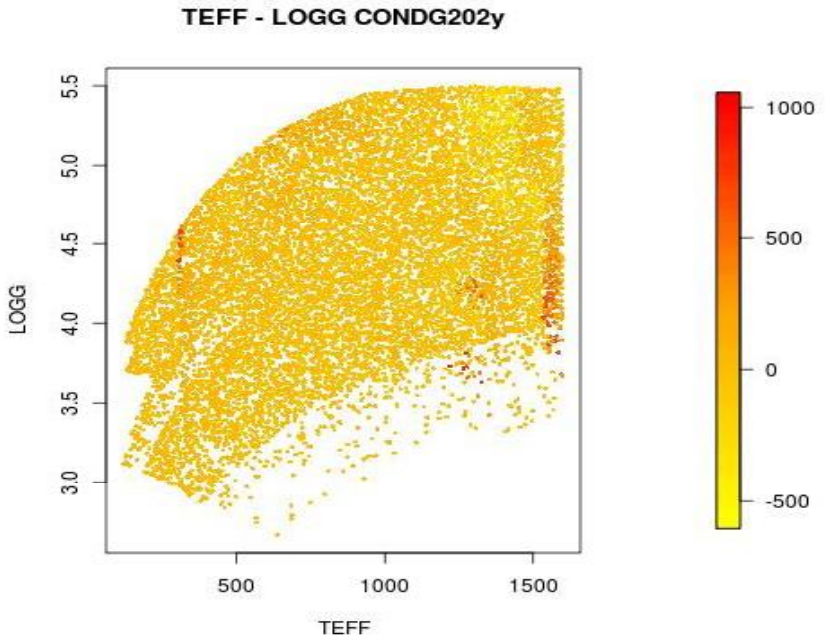


Figura 99. Gráfica de dispersión Teff vs Logg sobre PCA−KNN, para el conjunto de espectros CONDG202Yareal

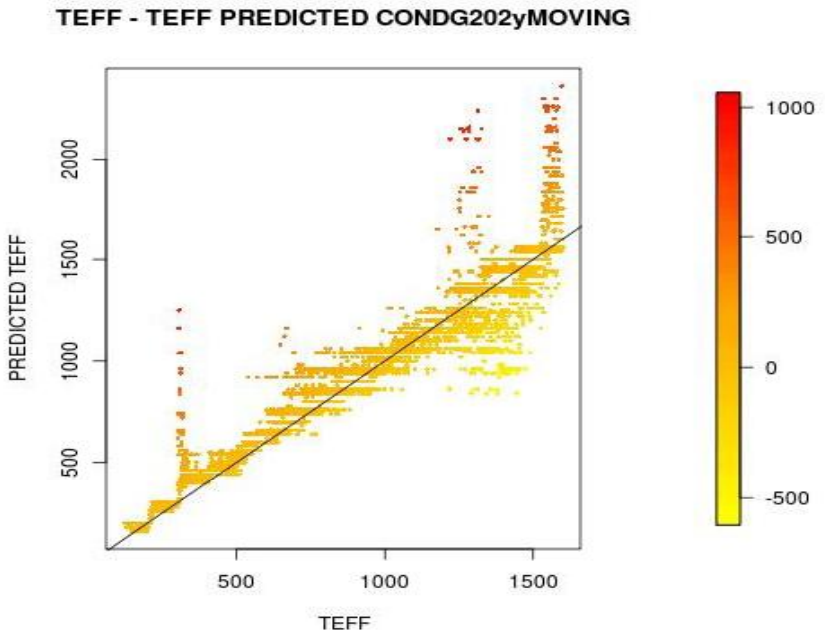


Figura 100 Gráfica de dispersión Teff vs Teff predicted, sobre PCA+KNN para el conjunto CONDG202Yarealmoving.

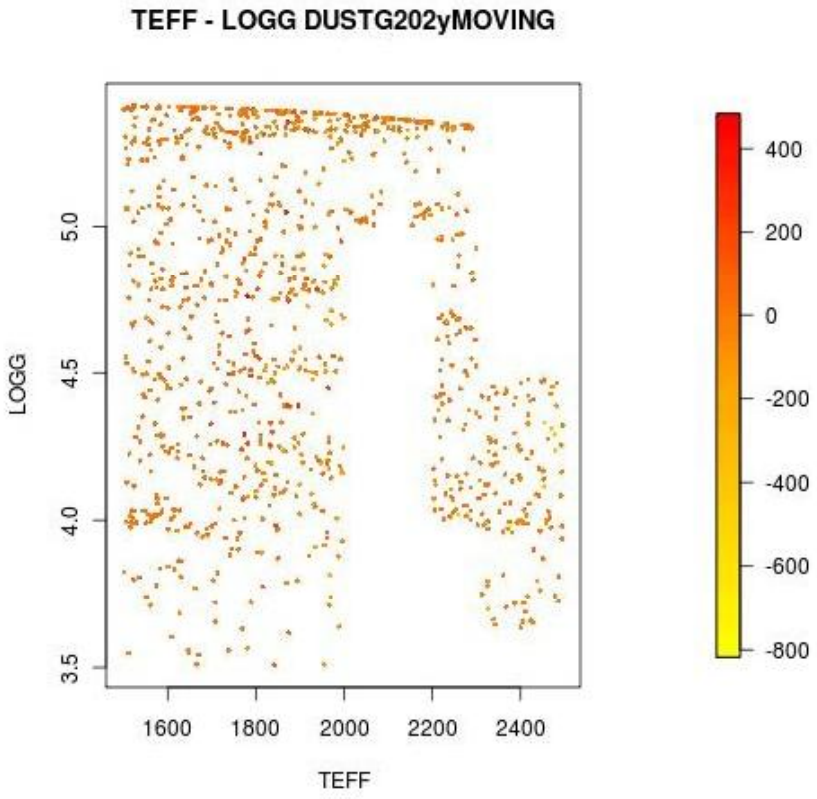


Figura 101. Gráfica de dispersión Teff vs Logg sobre PCA+KNN, para el conjunto de espectros CONDG20area1moving

Se observa como para el conjunto de espectros CONDG202Yarea1, la aplicación del suavizado mejora los resultados, tanto en el error medio como en los valores máximos de error

Sigue observándose el escalonado sobre las temperaturas inferiores a 500 º Kelvin

Los errores en la predicción son inasumibles, los problemas presentados con los conjuntos de espectros CONDG20 se multiplican en fases intermedias de la misión.

Para modelos COND, este clasificador se descarta completamente.

A continuación, la tabla 27 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG202Yarea1 y DUSTG202Yarea1moving

| | DUSTG202Y | DUSTG202Y moving |
|---|---|---|
| Correlation coefficient | 0.9502 | 0.9534 |
| Mean absolute error | 57.9685 | 54.9872 |
| Root mean squared error | 87.7645 | 84.4117 |
| Relative absolute error | 8.5164 % | 8.0784 % |
| Root relative squared error | 11.9401 % | 11.5221 % |

Tabla 27: Resultados para PCA+KNN de los conjuntos de espectros DUSTG202Yarea1, DUSTG202Yarea1moving

Las figuras 102 y 104 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador PCA+KNN.

Las figuras 103 y 105 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador PCA+KNN.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

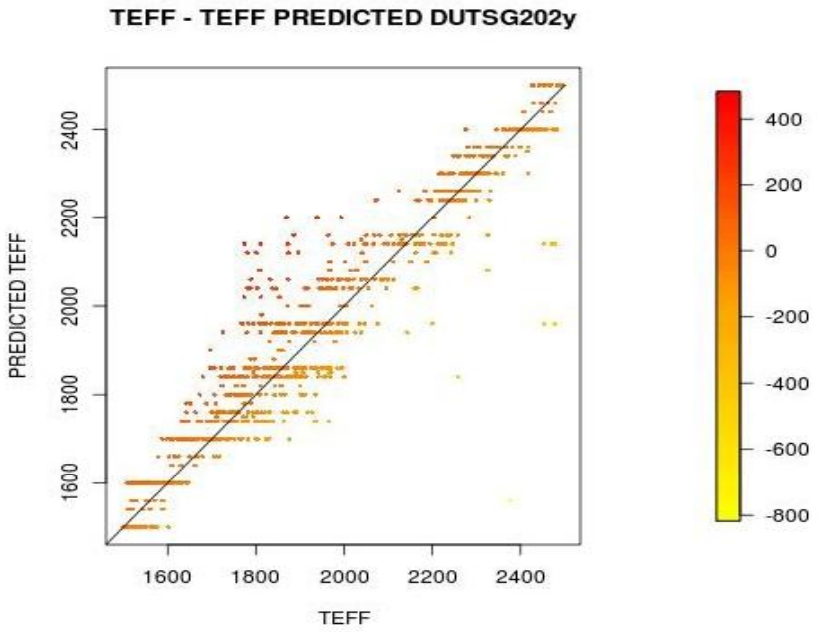


Figura 102 Gráfica de dispersión TeffI vs Teff predicted, sobre PCA KNN para el conjunto de espectros DUSTG202Yarea1.

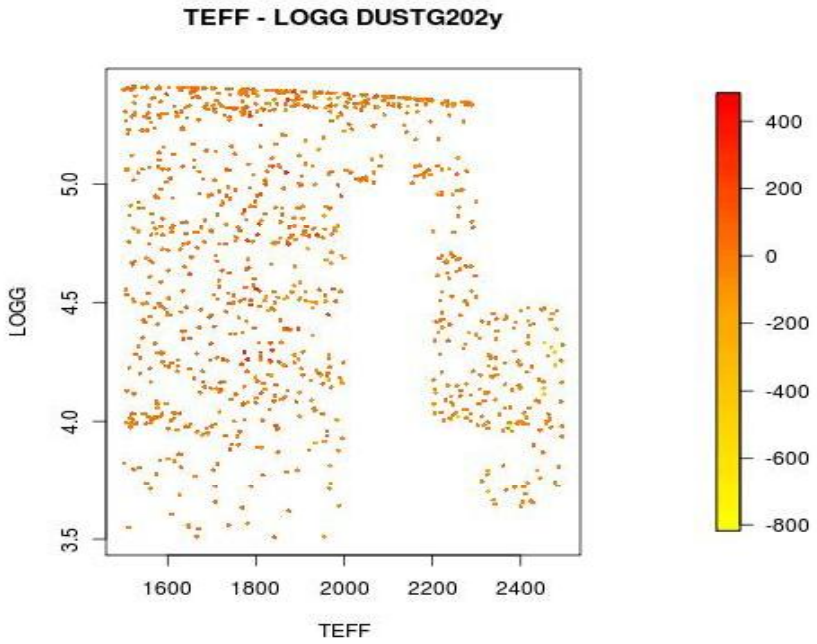


Figura. 103. Gráfica de dispersión TeffI vs Logg sobre PCA KNN. para el conjunto de espectros DUSTG202Yarea1

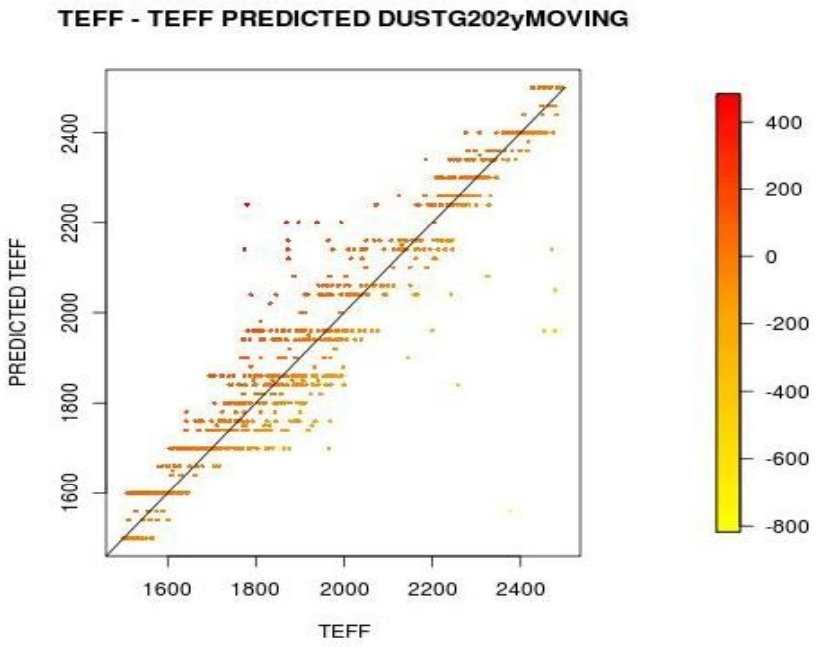


Figura 104 Gráfica de dispersión Teff vs Teff predicted, sobre PCA KNN para el conjunto DUSTG20area1moving.

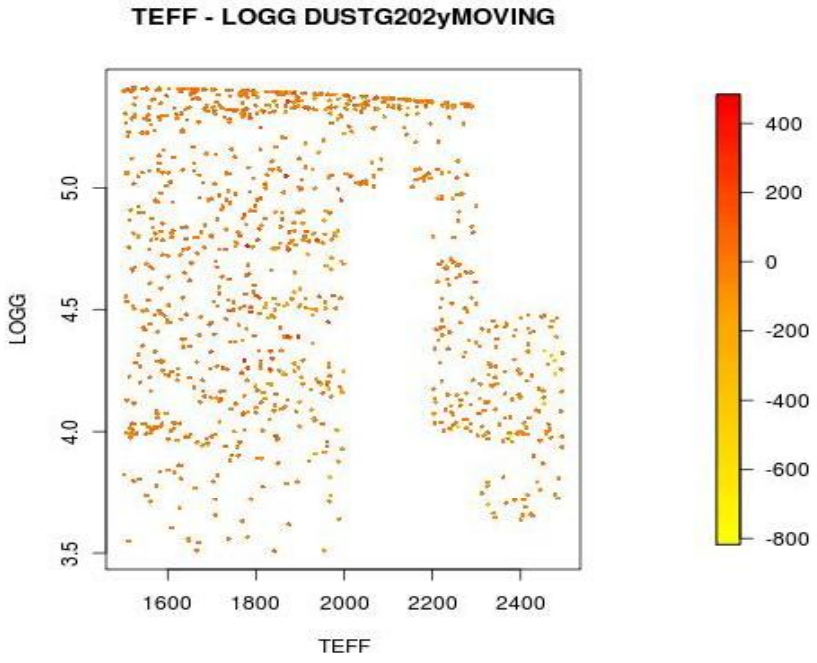


Figura 105 Gráfica de dispersión TeF vs Logg sobre PCA KNN, para el conjunto de espectros DUSTG20area1moving

Los conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación, por lo tanto, es normal que los resultados obtenidos sean peores que los conjuntos de datos de validación DUSTG20area1

Para el conjunto de espectros DUSTG202Yarea1, el suavizado le afecta muy poco de forma que aunque por la información proporcionada se podría decir que la predicción mejora, haría falta aplicar otros métodos para determinar que clasificador ofrece mejores resultados.

Sin embargo, este clasificador presenta comentado con anterioridad, existen determinadas temperaturas que predice muy mal de forma sobre una temperatura devuelta por el clasificador, no existe un margen de confianza para poder estimar el error existente sobre esa predicción.

Observando las figuras 102, 103, 104 y 105 no se pueden extraer conclusiones que nos determinen algún rango de temperaturas sobre las cuales las predicciones sean mejores

#### 3.2.2.1.2 Resultados para Máquinas de Vectores Soporte

Para la aplicación de Máquinas de Vectores Soporte previa reducción y transformación de los conjuntos de entrenamiento y validación se ha empleado Weka, mediante el clasificador weka.classifiers.functions.SMOreg, teniendo en cuenta el uso del nucleo RBF.

Tras diferentes experimentos de optimización de parámetros se termina que el modelo predictor óptimo emplea margen blando con valor 50,000 y un factor Gamma del nucleo RBF con valor 1,000 para el entrenamiento con el conjunto de entrenamiento de NOM y la validación con el conjunto de entrenamiento de RANG20.

Estos valores tan altos hacen que la ejecución, búsqueda y determinación de parámetros haya sido muy costosa temporalmente

La Tabla 28 muestra los resultados obtenidos tanto para validación cruzada como para la validación con RANG15area1

| | NOMarea1 | RANG15 |
|---|---|---|
| Correlation coefficient | 0.9974 | 0.994 |
| Mean absolute error | 21.7785 | 33.0559 |
| Root mean squared error | 49.0559 | 56.2177 |

Tabla 28: Resultados sobre PCA_SMO para validación cruzada y para el conjunto de validación sin ruido RANG15area1.

Nos apoyamos en las gráficas de dispersión sobre la validación con RANG15area1 (figuras 106 y 107) para extraer conclusiones. Téngase en cuenta que el degradado de color muestra el error en la predicción de la temperatura con respecto a la real

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

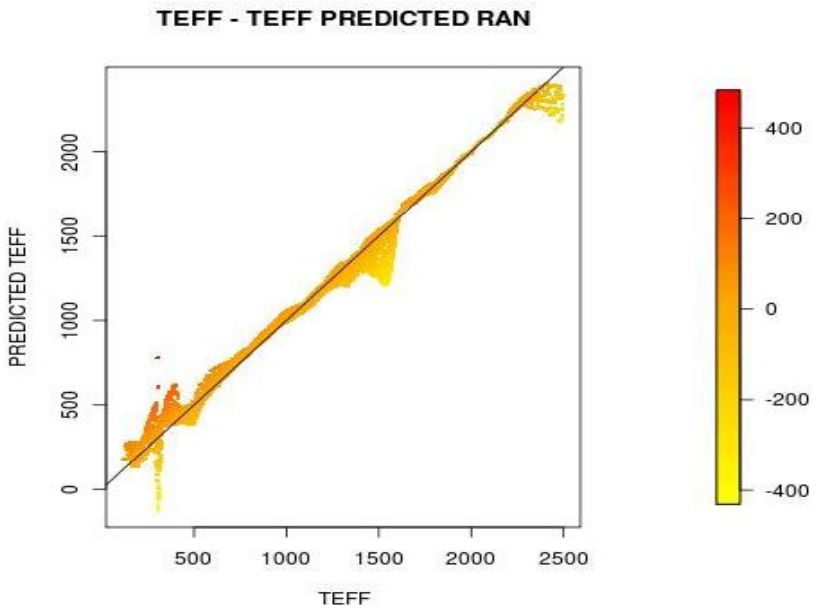


Figura 106 Gráfica de dispersión para la predicción sobre PCA1SMO de TEFF vs TEFF predicted para el conjunto de validación RANareal

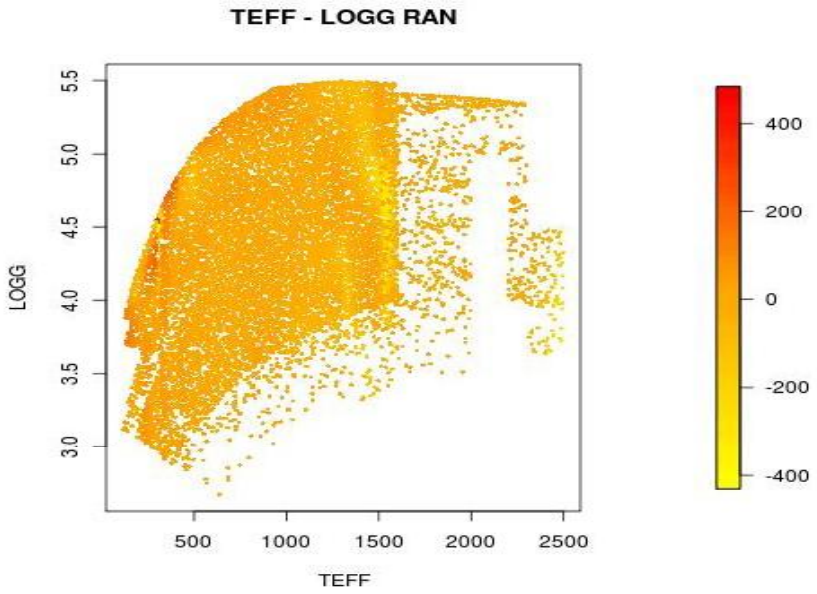


Figura 107 – Gráfica de dispersión para la predicción sobre PCA ISMO de TEFF vs LOG para el conjunto de validación RANarea1.

Para este clasificador, se observa tanto en la figura 106 como 107 que, las mejores predicciones se encuentran para el rango de temperatura entre 600 y 1200 grados Kelvin y el rango entre 1600º y 2200 º Kelvin.

Un dato a observar es que en el rango entre 600 y 1000 º Kelvin, el sistema predice temperaturas por encima de las reales

Es interesante observar como para las temperaturas más elevadas en los modelos COND (1600º K) y DUST (2500º K) el clasificador precide muy mal. Debería estudiarse con mayor detenimiento si el problema viene determinado por algún error en el clasificador o en la transformación de los atributos.

También se observa que por debajo de 500º Kelvin el clasificador también tiene un extraño comportamiento que podría deberse a los errores comentados en los clasificadores anteriores

A continuación, en un estudio más profundo se va a reevaluar el clasificador para los conjuntos de espectros con ruido comentados en la tabla 7

La tabla 29, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG20areal y CONDG20arealmoving

| | CONDG20 | CONDG20 moving |
|---|---|---|
| Correlation coefficient | 0.9896 | 0.9856 |
| Mean absolute error | 38.5972 | 54.1168 |
| Root mean squared error | 61.2521 | 73.5347 |

Tabla 29: Resultados para PCA_SMO de los conjuntos de espectros CONDG20areal y CONDG20arealmoving

Las figuras 108 y 110 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG20areal y CONDG20arealmoving empleando el clasificador PCA+SMO

Las figuras 109 y 111 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG20areal y CONDG20arealmoving empleando el clasificador PCA+SMO.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

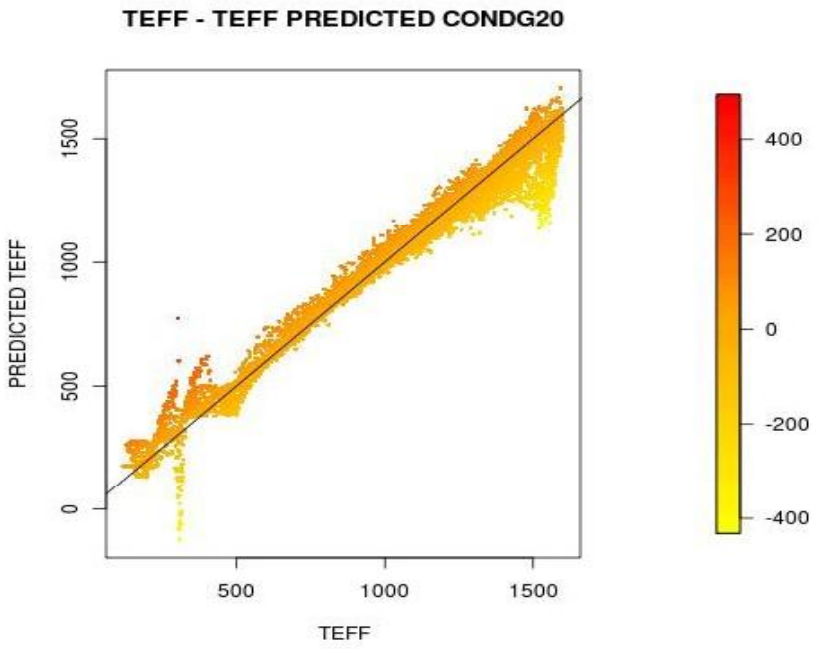


Figura 109 Gráfica de dispersión Teff vs Teff predicted, sobre PCA SMO para el conjunto de espectros CONDG20area1.

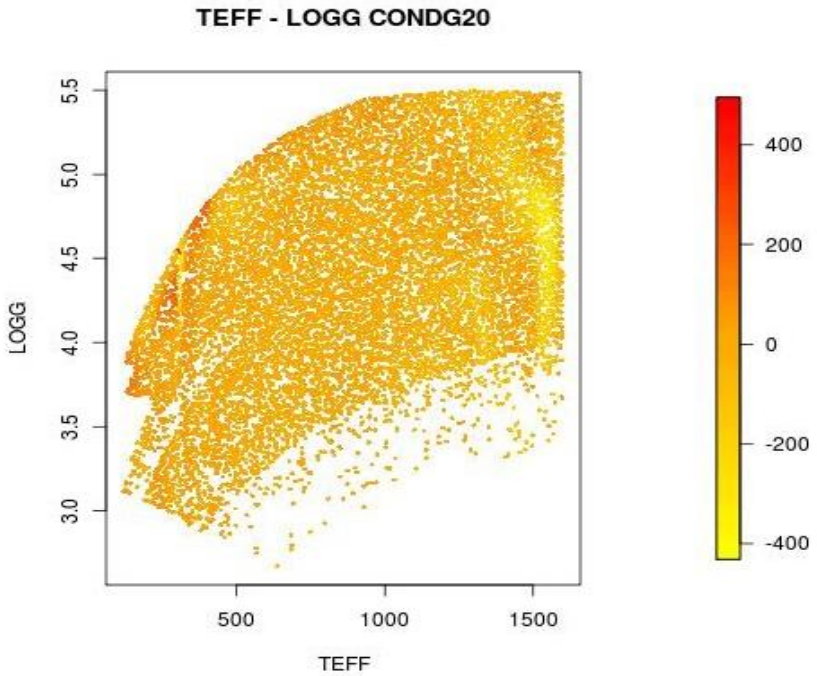


Figura 110. Gráfica de dispersión Teff vs Logg sobre PCA SMO, para el conjunto de espectros CONDG20area1

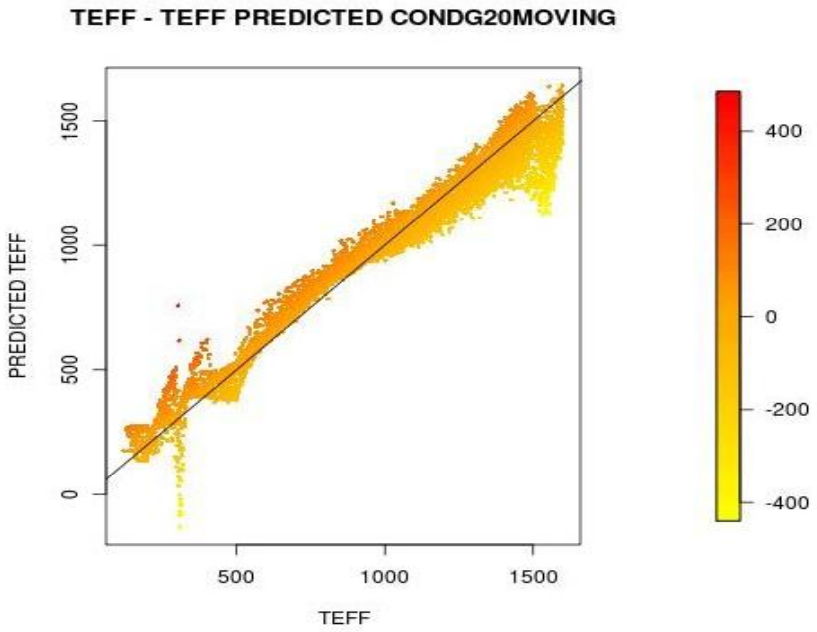


Figura. 111 Grafica de dispersión TeIf vs TeIf predicted, sobre PCA ISMO para el conjunto CONDG20area1moving.

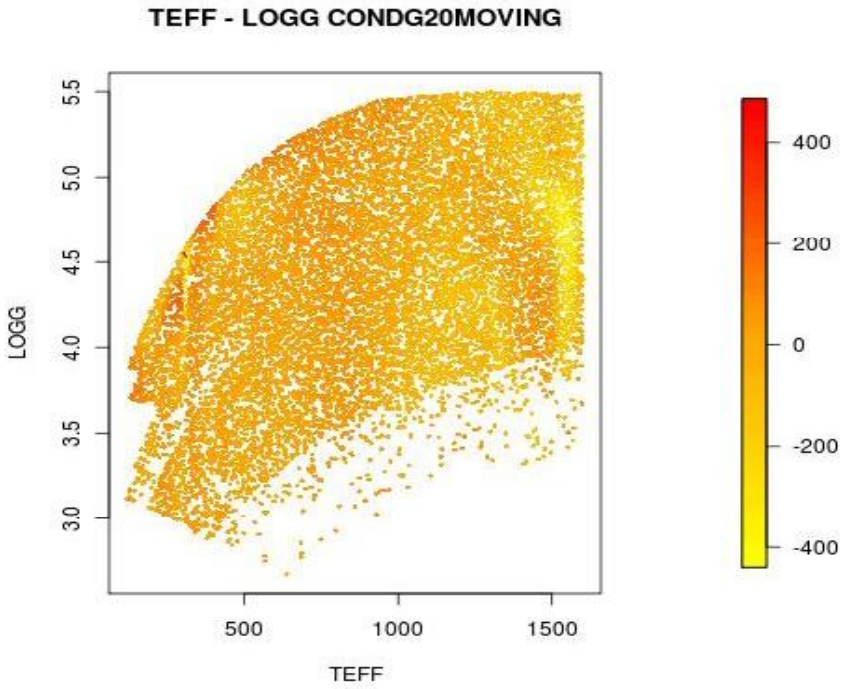


Figura 112. Grafica de dispersión TeIf vs Logg sobre PCA−SMO, para el conjunto de espectros CONDG20area1moving

Por un lado, observamos como para el conjunto de espectros CONDG20area1, la aplicación del suavizado no mejora los resultados.

Ocurren los mismos problemas detectados para los conjuntos de datos de validación sin ruido RANGI5area1:

- Para este clasificador, se observa que las mejores predicciones se encuentran para el rango de temperatura entre 600 y 1200 grados Kelvin.

- Un dato a observar es que en el rango entre 600 y 1000 ° Kelvin, el sistema predice temperaturas por encima de las reales

- Es interesante observar como para las temperaturas más elevadas en los modelos COND (1600° Kelvin) el clasificador procide muy mal

- También se observa que por debajo de 500° Kelvin el clasificador también tiene un extraño comportamiento que podría deberse a los errores visualizados en gráficas anteriores

A continuación, la tabla 30 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG20area1 y DUSTG20area1moving:

| | DUSTG20 | DUSTG20 moving |
|---|---|---|
| Correlation coefficient | 0.9787 | 0.9802 |
| Mean absolute error | 40.4993 | 45.1335 |
| Root mean squared error | 58.2473 | 62.6355 |

Tabla 30: Resultados para PCA_SMO de los conjuntos de espectros DUSTG20area1 y DUSTG20area1moving

Las figuras 113 y 115 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos DUSTG20area1 y DUSTG20area1moving empleando el clasificador PCA+SMO.

Las figuras 114 y 116 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos para validación DUSTG20area1 y DUSTG20area1moving empleando el clasificador SMO.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

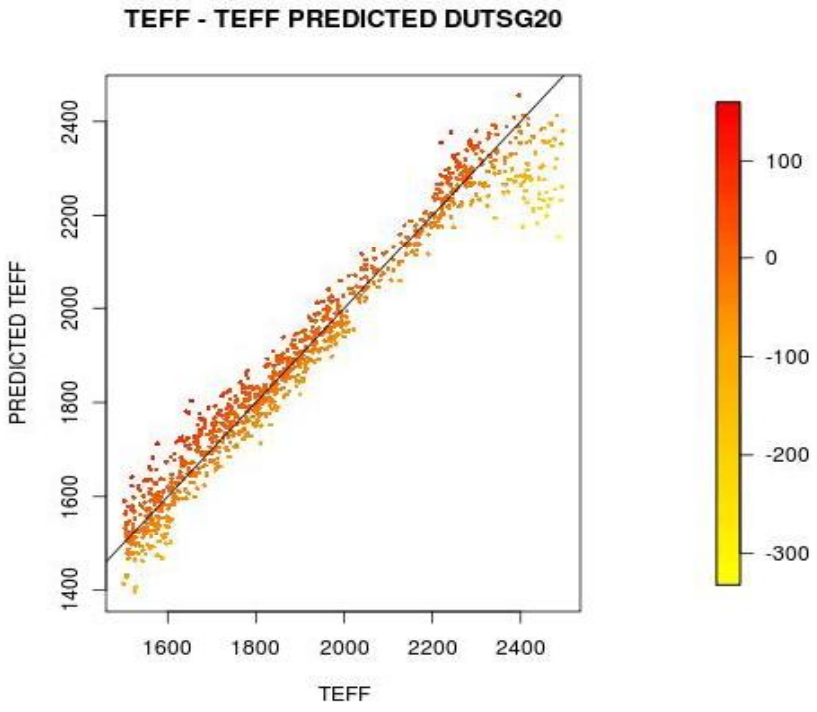


Figura 113 Grafica de dispersión Teff vs Teff predicted, sobre SMO para el conjunto de espectros DUSTG20area1.

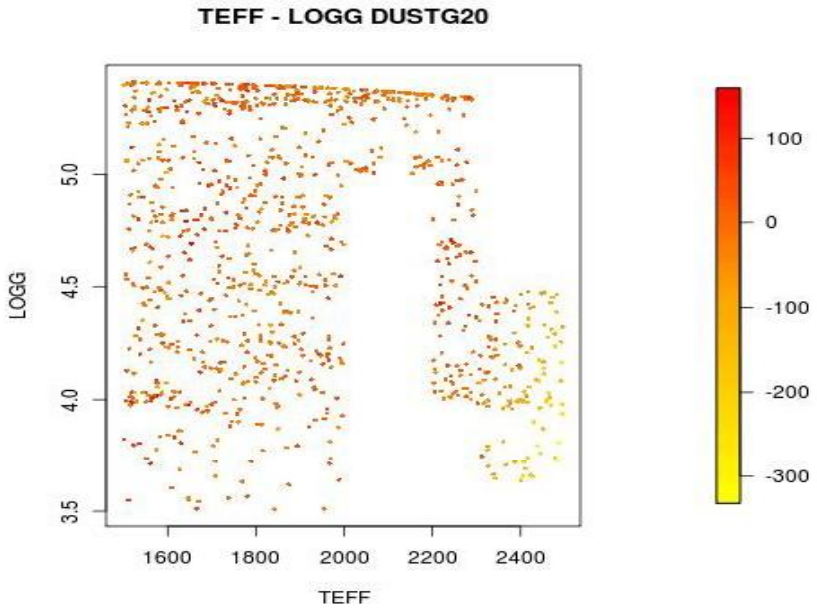


Figura 114. Gráfica de dispersión Teff vs Logg sobre SMO, para el conjunto de espectros DUSTG20area1

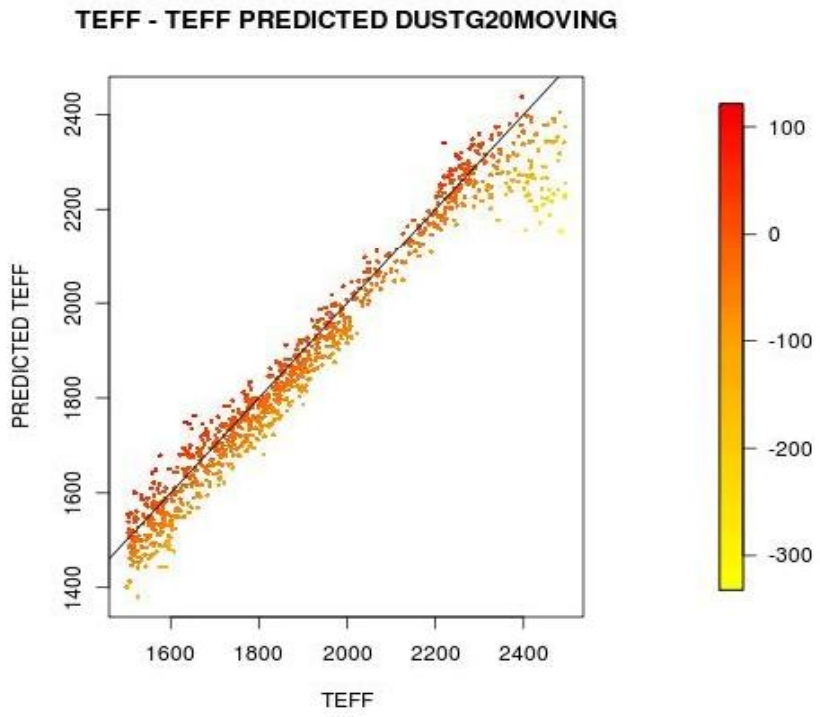


Figura 115 Gráfica de dispersión Teff vs Teff predicted, sobre PCA−SMO para el conjunto DUSTG20area1moving.

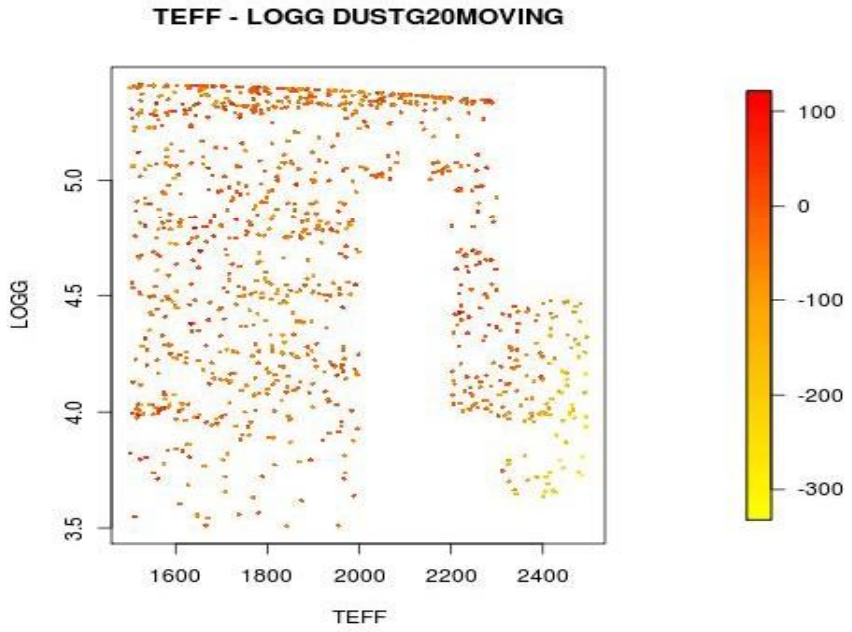


Figura 116. Grafica de dispersión TeFF vs Logg sobre PCA+SMO, para el conjunto de espectros DUSTG20area1moving

Por un lado, se observamos como para el conjunto de espectros DUSTG20area1 la aplicación del suavizado no supone un añadido, ya que el resultado es practicamente igual y las diferencias no se pueden apreciar con la información disponible

Para este clasificador, se observa tanto en la figura 114 como 116 que, las mejores predicciones se encuentran para el rango de temperatura entre 1600° y 2200 ° Kelvin

La mayoria de los errores en predicciones por encima de 2200° Kelvin son predicciones por debajo de la temperatura efectiva real.

Es interesante observar como para el rango final de temperatura (las temperaturas más elevadas en el modelo DUST (2500° Kelvin) el clasificador preside muy mal Es posible que haya algún problema en el clasificador para el final de rango, o algún problema en la aplicación de transformación PCA, ya que este mismo problema ocurre con el fin de rango para el modelo COND.

A continuación, se presenta un estudio del clasificador SMO sin reducción de dimensionalidad, para los conjuntos de validación CONDG202Y y DUSTG202Y, es decir, sobre conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación.

La tabla 31, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG202Yareal y CONDG202Yarealmoving

| | CONDG202Y | CONDG202Y moving |
|---|---|---|
| Correlation coefficient | 0.9418 | 0.9385 |
| Mean absolute error | 92.6032 | 96.1995 |
| Root mean squared error | 135.0973 | 137.9925 |

Tabla 31: Resultados para PCA+SMO de los conjuntos de espectros CONDG202Yareal

Las figuras117 y 119 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG202Yareal y CONDG202Yarealmoving empleando el clasificador PCA+SMO.

Las figuras 118 y 120 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG202Yareal y CONDG202Yarealmoving empleando el clasificador PCA+SMO

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

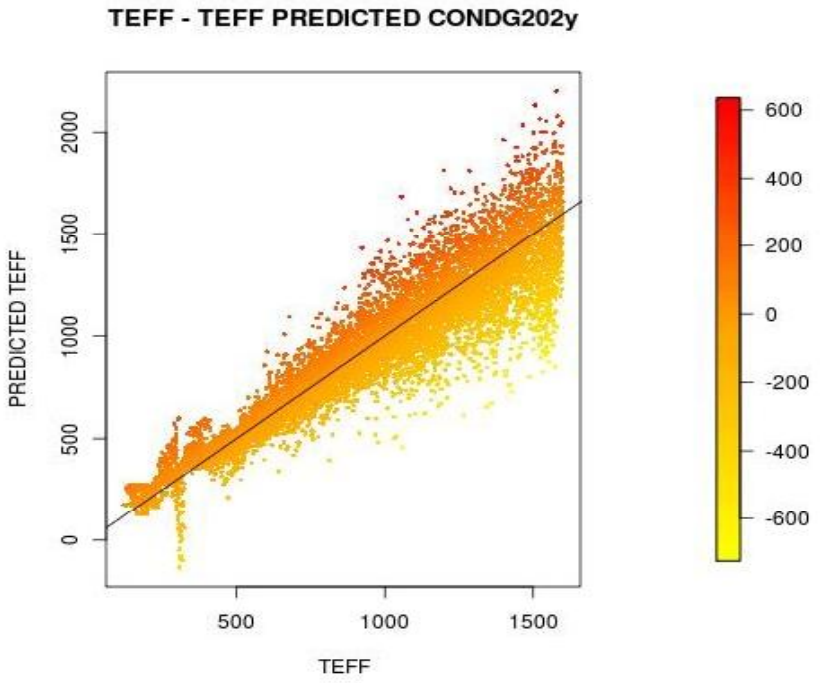


Figura 117 Gralica de dispersión: TeIl' vs TeIl'medicted, sobre PCA1SMO para el coujunto de espectros CONDG202Yreal.

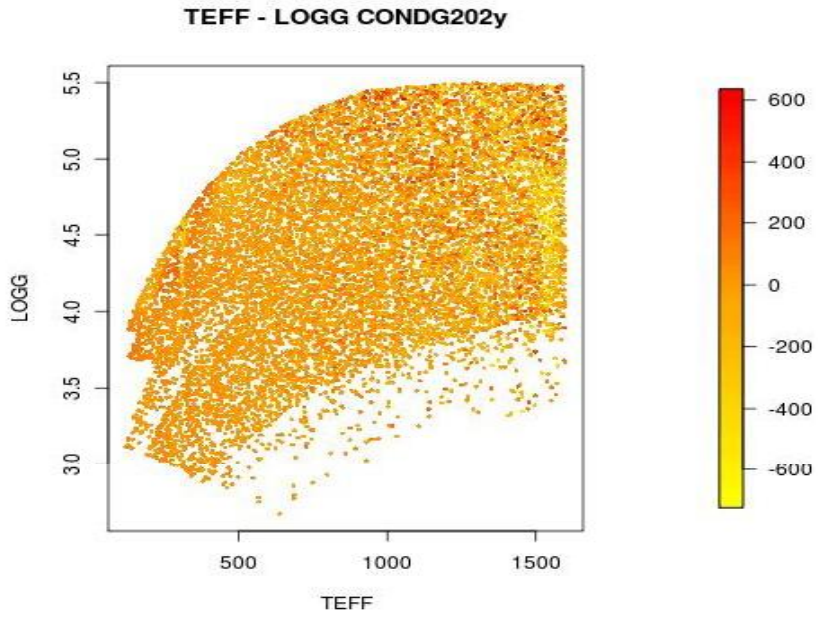


Figura 118. Gralica de dispersión TeIl' vs Logg sobre PCA1SMO, para el coujunto de espectros CONDG202Yreal

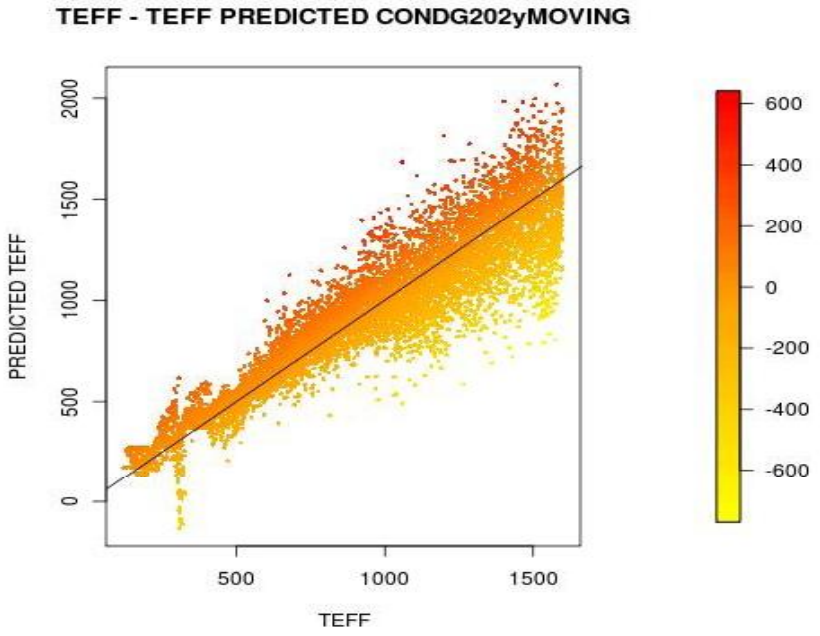


Figura 119 Gráfica de dispersión Teff vs Teff predicted, sobre PCA−SMO para el conjunto CONDG202Yarea1moving.

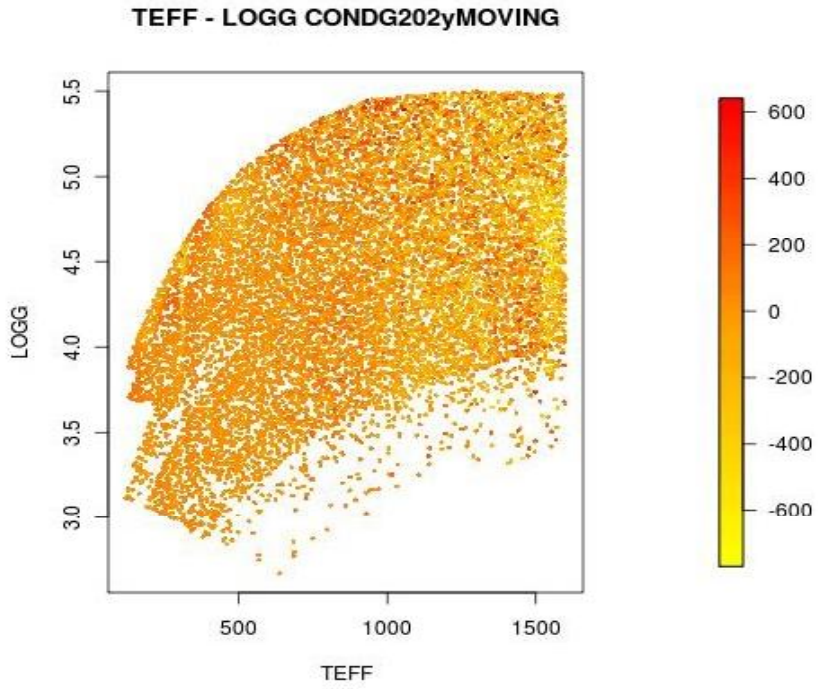


Figura 120. Gráfica de dispersión Teff vs Logg sobre PCA1SMO, para el conjunto de espectros CONDG20area1moving

Se observa como para el conjunto de espectros CONDG202Yarea1, la aplicación del suavizado no mejora los resultados

Ocurren los mismos problemas detectados para los conjuntos de datos de validación sin ruido RANGI5area1:

- Para este clasificador, se observa que las mejores predicciones se encuentran para el rango de temperatura entre 600 y 1200 grados Kelvin.

- Es interesante observar como para las temperaturas más elevadas en los modelos COND (1600º Kelvin) el clasificador procide muy mal

Sin embargo los errores en predicción son tan elevados que hacen inviable considerar este clasificador sobre los modelos COND. para etapas intermedias de la misión

A continuación, la tabla 32 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG202Yarea1 y DUSTG202Yarea1moving

| | DUSTG202Y | DUSTG202Y moving |
|---|---|---|
| Correlation coefficient | 0.7683 | 0.7841 |
| Mean absolute error | 158.7818 | 157.5563 |
| Root mean squared error | 205.142 | 201.5095 |

Tabla 32: Resultados para SMO de los conjuntos de espectros DUSTG202Yarea1, DUSTG202Yarea1moving

Las figuras 121 y 123 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador SMO

Las figuras 122 y 124 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador SMO.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

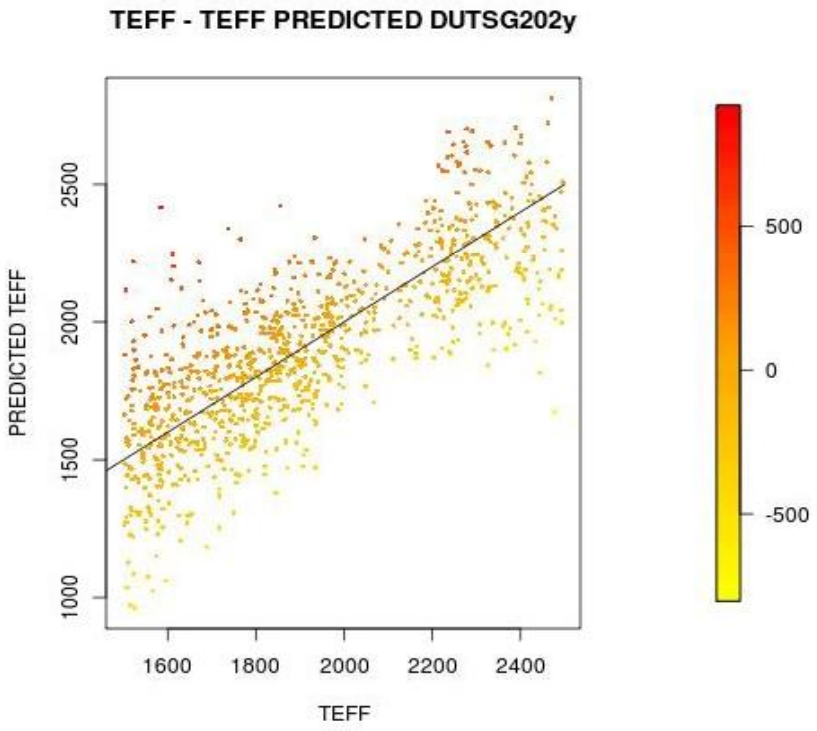


Figura 131 Grafica de dispersión Teff vs Teff predicted, sobre PCA+SMO para el conjunto de espectros DUSTG202Yarea1.

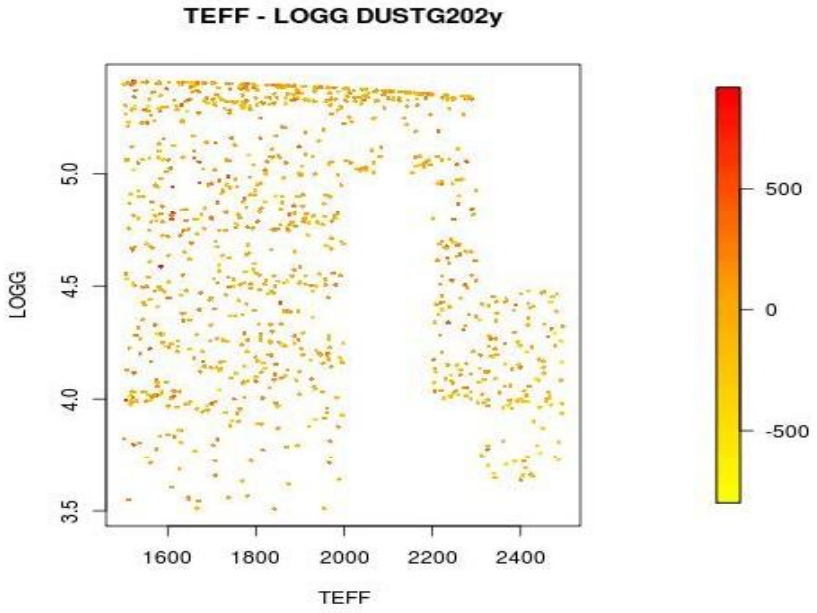


Figura 122. Gráfica de dispersión TeffVs Logg sobre PCA SMO. para el conjunto de espectros DUSTG202Yreal

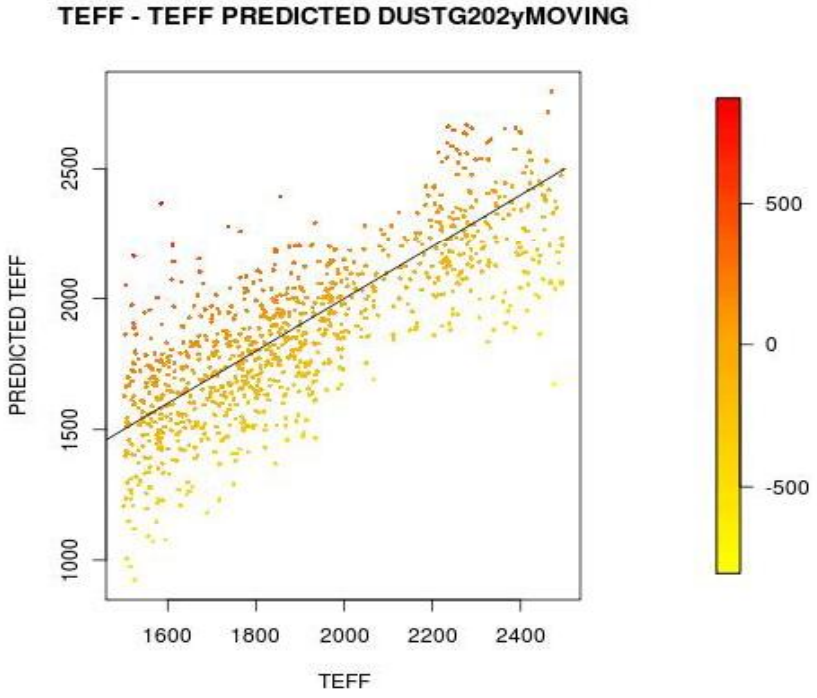


Figura 123 Gráfica de dispersión Teff vs Teff predicted, sobre PCA SMO para el conjunto DUSTG20Yareal moving.

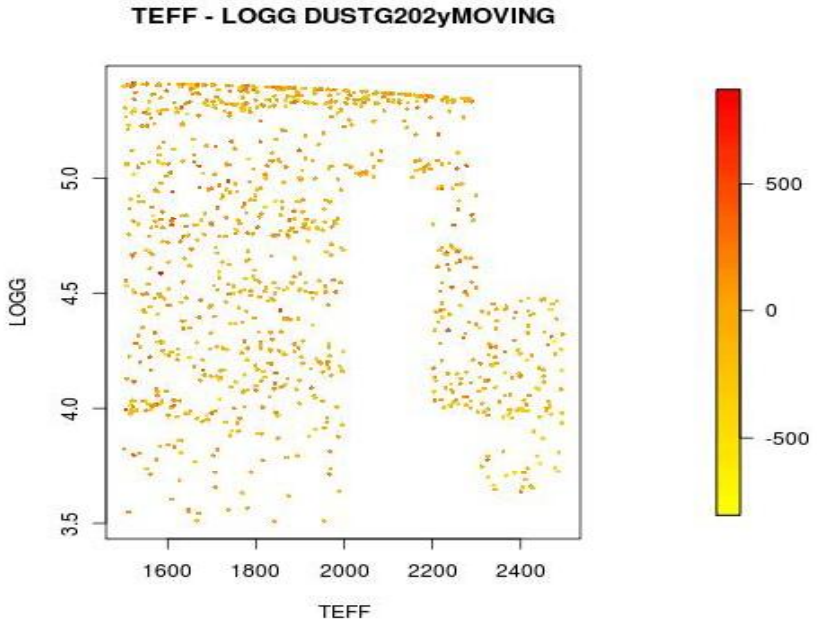


Figura 124. Gráfica de dispersión Teff vs Logg sobre PCA_SMO, para el conjunto de espectros DUSTG20area1moving

Los conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación, por lo tanto, es normal que los resultados obtenidos sean peores que los conjuntos de datos de validación DUSTG20area1

Observando las figuras 121, 122, 123 y 124 no se pueden extraer conclusiones que nos determinen algún rango de temperaturas sobre las cuales las predicciones sean mejores

Los resultados obtenidos de la aplicación de Máquinas de Vectores Soporte sobre los conjuntos de datos transformados previamente por PCA, no aportan mejores resultados que los obtenidos con los anteriores clasificadores

Para CONDarea1 para magnitud aparente G20, los errores no son controlados de forma que no hay ninguna zona donde la predicción tenga unos márgenes de error razonables.

Para etapas intermedias de la misión, los errores en predicción son tan elevados que hacen inviable

considerar este clasificador

#### 3.2.2.1.3 Resultados para Procesos Gausianos

Para la aplicación de Procesos Gausianos sobre los datos en PCA analizados en el apartado 3.2.2.1, se ha empleado el algoritmo de weka weka.classifiers.functions.GaussianProcesses, teniendo en cuenta el uso del nucleo RBF. Los principales parámetros a ajustar en weka para la optimización del algoritmo predictor son dos:

- Noise – Nos determina el nivel del Ruido Gausiano (el cual es añadido la diagonal de la Matriz de Covarianza)

- Gamma – El factor Gamma del nucleo RBF.

Tras diferentes experimentos de optimización de parámetros se termina que el modelo predictor óptimo emplea un nivel de ruido Gausiano igual a 0,08 y un factor Gamma del nuclero RBF con valor 10 para el entrenamiento con el conjunto de entrenamiento de NOM y la validación con el conjunto de entrenamiento de RANG20.

La Tabla 33 muestra los resultados obtenidos tanto para validación cruzada como para la validación con RANG15area1

| | NOMarea1 | RANG15 |
|---|---|---|
| Correlation coefficient | 0.999 | 0.9939 |
| Mean absolute error | 9.9069 | 34.465 |
| Root mean squared error | 31.419 | 54373 |
| Relative absolute error | 1.7174 % | 7.6191 % |
| Root relative squared error | 4.6269 % | 9.9759 % |

Tabla 33: Resultados sobre PCA+GPS para validación cruzada y para el conjunto de validación sin ruido RANG15area1.

Nos apoyamos en las gráficas de dispersión sobre la validación con RANG15area1 (figuras 125 y

126) para extraer conclusiones. Téngase en cuenta que el degradado de color muestra el error en la predicción de la temperatura con respecto a la real

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

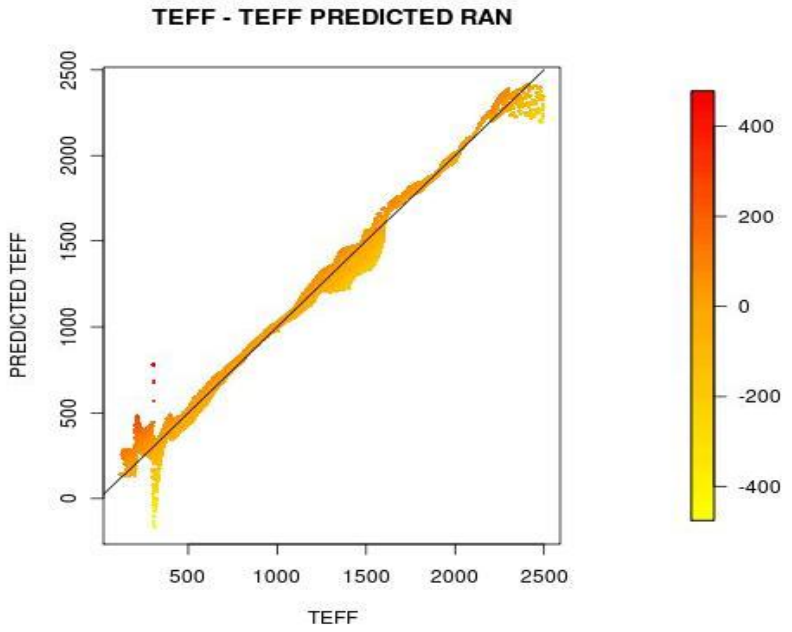


Figura 125 Gráfica de dispersión para la predicción sobre PCA GPS de TEFF vs TEFF predicted para el conjunto de validación RANreal

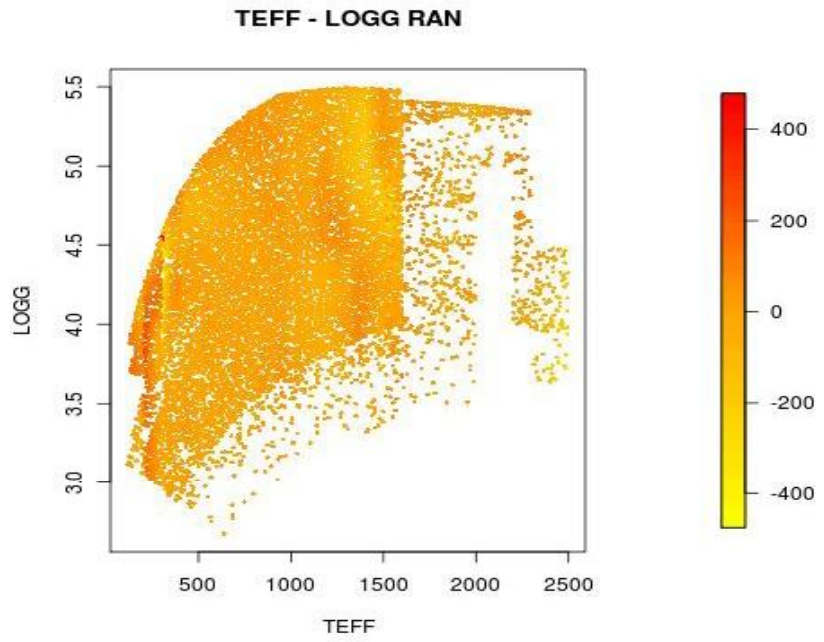


Figura 126 Gráfica de dispersión para la predicción sobre PCA 1GPS de TEFF vs LOG para el conjunto de validación RANarea1.

Los valores de temperatura efectiva real 400º Kelvin muestran un error muy variable que desvirtuan el comportamiento real del clasificador, este error es constante, para esta zona en los otros clasificadores que emplean la transformación PCA

El motivo de este error podría ser que al transformar a PCA se ha perdido información referente a esa temperatura.

Habría que considerar emplear un mayor número de componentes en el cálculo de PCA

Para este clasificador, se observa tanto en la figura 125 como 126 que, las mejores predicciones se encuentran para el rango de temperatura entre 500º y 1300º K y el rango entre 1700º y 2200º K.

Al igual que en el caso de los otros clasificadores, en temperaturas por encima de los 2200º K el

clasificador se vuelve muy inestable y predice temperaturas muy por debajo del valor real. Este comportamiento, al ser constante en los otros clasificadores, debe ser debido a la perdida de información al aplicar la transformada.

A continuación, en un estudio más profundo se va a reevaluar el clasificador para los conjuntos de espectros con ruido comentados en la tabla 7

La tabla 34, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG20area1 y CONDG20area1moving

| | CONDG20 | CONDG20 moving |
|---|---|---|
| Correlation coefficient | 0.99 | 0.9882 |
| Mean absolute error | 37.5065 | 46.2138 |
| Root mean squared error | 57.3031 | 64.109 |
| Relative absolute error | 8.7322 % | 10.7594 % |
| Root relative squared error | 10.9723 % | 12.2754 % |

Tabla 34: Resultados para PCA+GPS de los conjuntos de espectros CONDG20area1 y CONDG20area1moving

Las figuras 127 y 129 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG20area1 y CONDG20area1moving empleando el clasificador PCA+GPS

Las figuras 128 y 130 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG20area1 y CONDG20area1moving empleando el clasificador PCA+GPS.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos

de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

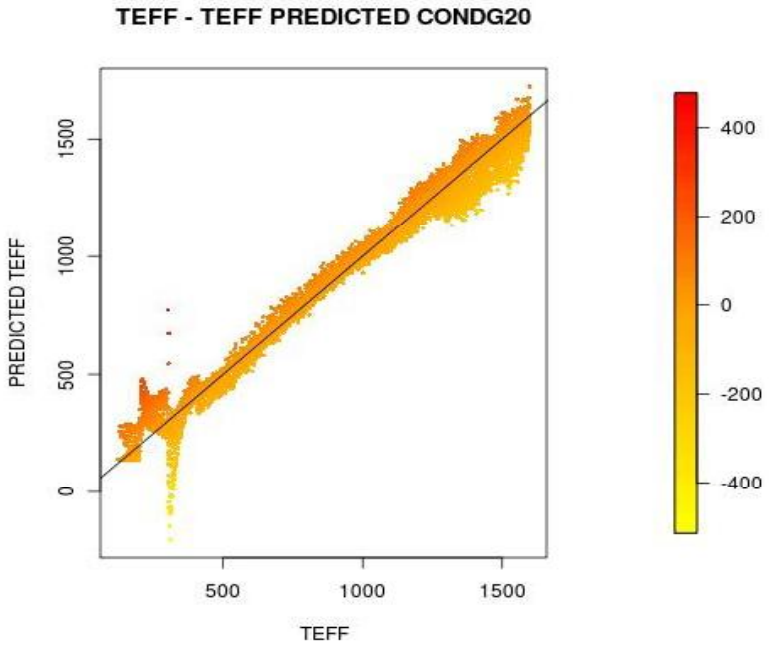


Figura 127 Gráfica de dispersión Teff vs Teff predicted, sobre PCA GPS para el conjunto de espectros CONDG20area1.

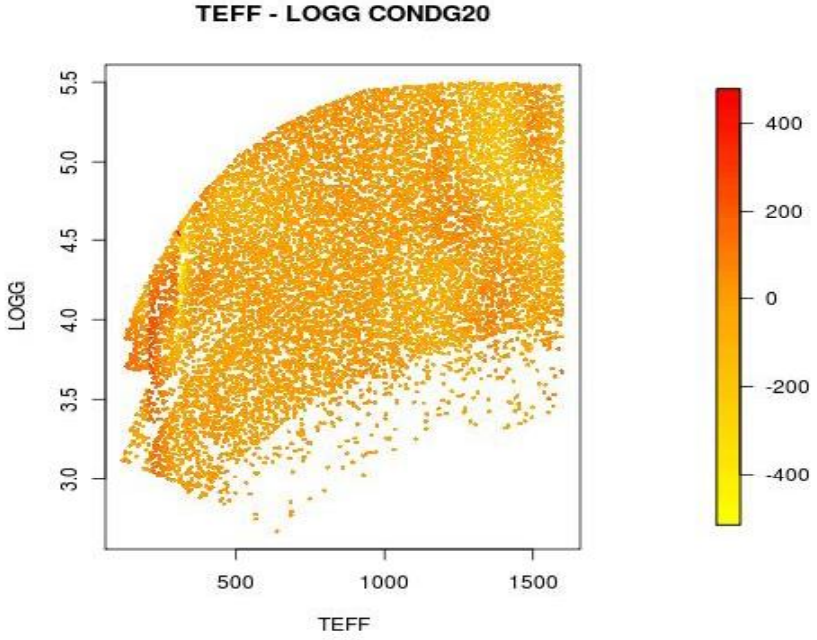

Figura 128. Grafica de dispersión TeffEvs Logg sobre PCA_GPS, para el conjunto de espectros CONDG20area1

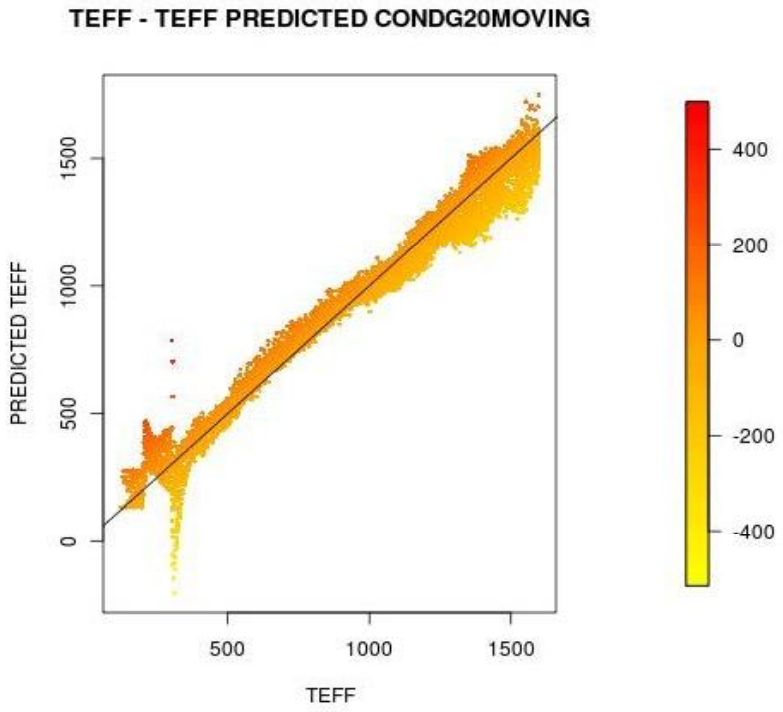


Figura 129. Grafica de dispersión TeffEvs TeffEpredicted, sobre PCA1GPS para el conjunto CONDG20area1moving.

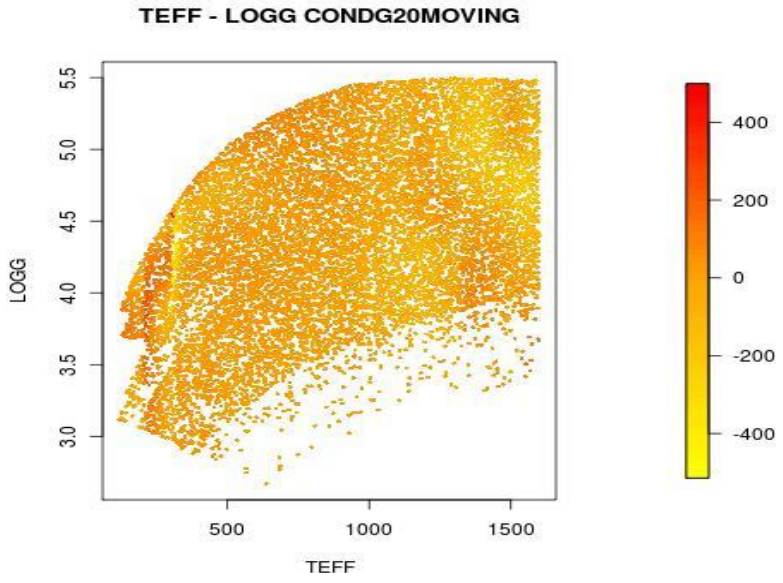


Figura 130. Grafica de dispersión TeffEvs Logg sobre PCA-GPS, para el conjunto de espectros CONDG20area1moving

Por un lado, observamos como, para el conjunto de espectros CONDG20area1, la aplicación del suavizado no solo no mejora los resultados, sino que incluso los empeora ampliando los valores de los errores máximos.

Ocurren los mismos problemas detectados para los conjuntos de datos de validación sin ruido RANG15area1:

- Entorno a los valores de temperatura efectiva real 400" Kelvin se sigue mostrando un error muy variable que desvirtua el comportamiento real del clasificador, este error es constante en los otros clasificadores que emplean la transformación PCA

  Para este rango de temperatura se producen los máximos errores con cifras de predicción por encima y por debajo de la temperatura efectiva real

El motivo podria ser que al transformar a PCA se ha perdido información referente a esa temperatura, ya que este error es repetitivo en los otros clasificadores que usan PCA.

Para este clasificador, se observa tanto en la figura 127 como 129 que, las mejores predicciones se encuentran para el rango de temperatura entre 500" y 1300" K.

A continuación, la tabla 35 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG20area1 y DUSTG20area1moving:

| | DUSTG20 | DUSTG20 moving |
|---|---|---|
| Correlation coefficient | 0.9759 | 0.9776 |
| Mean absolute error | 45.0704 | 43.0333 |
| Root mean squared error | 61.639 | 58.7785 |
| Relative absolute error | 6.6215 % | 6.3222 % |
| Root relative squared error | 8.3858 % | 7.9967 % |

Tabla 35: Resultados para PCA GPS de los conjuntos de espectros DUSTG20area1 y DUSTG20area1moving

Las figuras 131 y 133 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos DUSTG20area1 y

DUSTG20areal moving empleando el clasificador PCA+GPS.

Las figuras 132 y 134 nos muestran respectivamente gráficas de dispersión de temperatura real frente a el logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos para validación DUSTG20areal y DUSTG20areal moving empleando el clasificador PCA−GPS

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

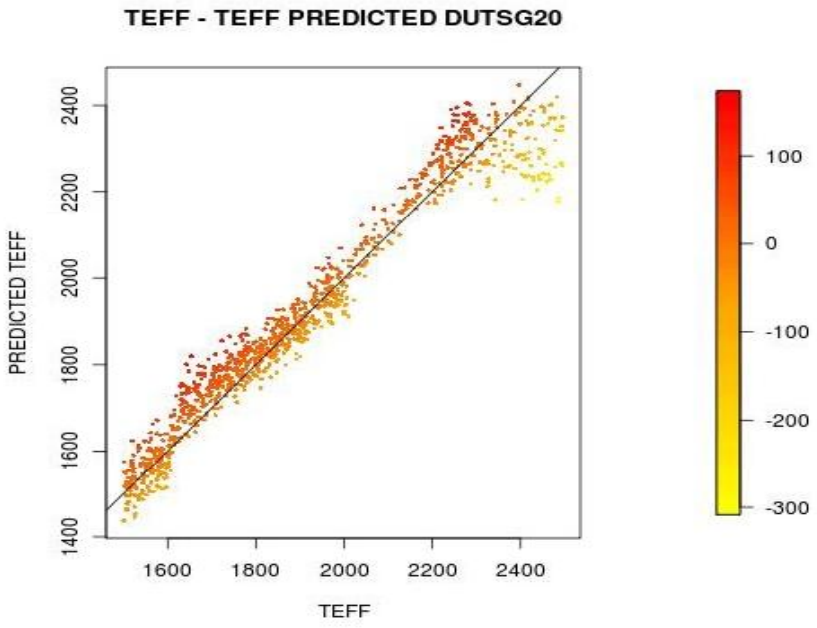


Figura 131 Gráfica de dispersión Teff vs Teff predicted, sobre PCA GPS para el conjunto de espectros DUSTG20areal.

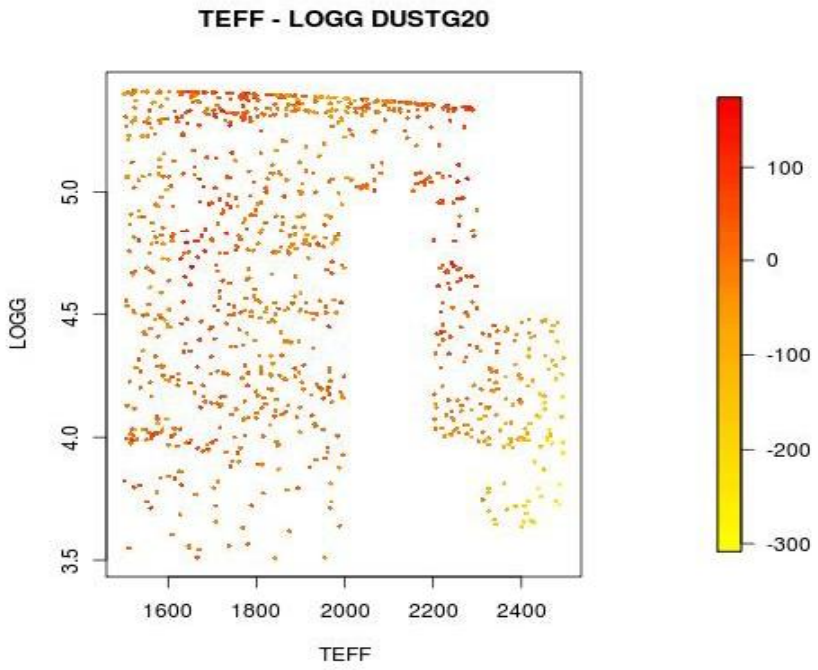


Figura 132. Gráfica de dispersión Teff vs Logg sobre PCA1GPS, para el conjunto de espectros DUSTG20area1

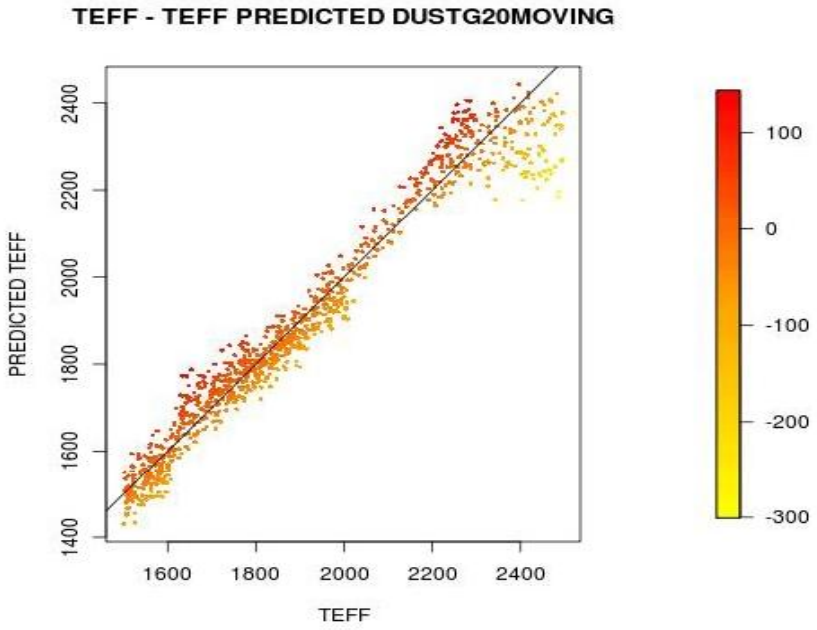


Figura 133 Gráfica de dispersión Teff vs Teff predicted, sobre PCA-GPS para el conjunto DUSTG20area1moving.

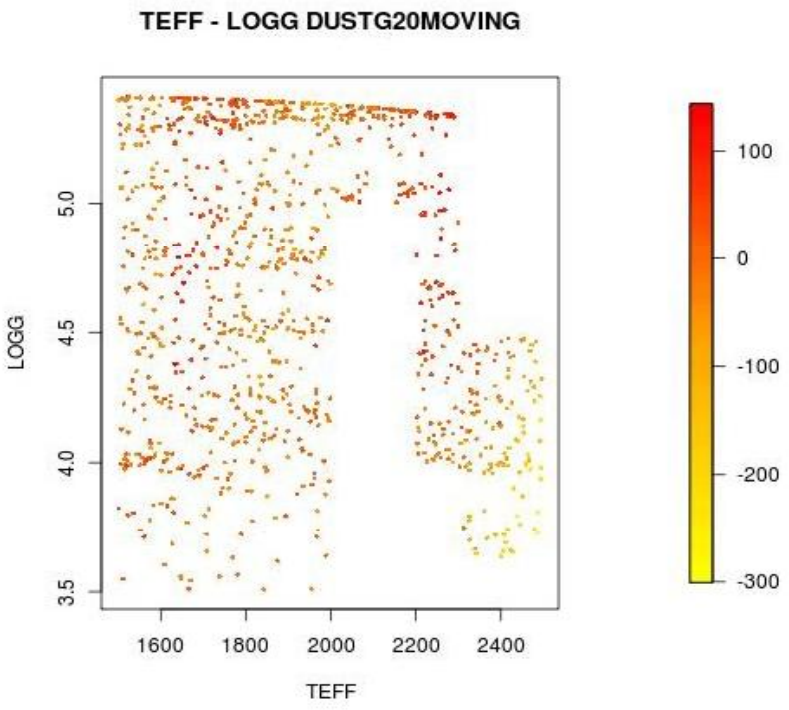


Figura 134 Grafica de dispersión Teff vs Logg sobre PCA+GPS, para el conjunto de espectros DUSTG20area1moving

Por un lado, se observamos como para el conjunto de espectros DUSTG20area1 la aplicación del suavizado existe una pequeña diferencia pero no es muy apreciable, para poder determinar si el clasificador realmente mejora, deberían aplicarse sistemas de Inferencias Bayesianas o T-Student

Por los resultados anticipados sobre RANGE5area1, y teniendo en cuenta que los modelos DUST son espectros con temperatura efectiva superior a 1500 grados Kelvin, las predicciones para temperatura real entre 1600º Kelvin hasta 2200º Kelvin ofrecen muy buenas predicciones.

Por encima de los 2200º Kelvin, para los modelos DUST, el clasificador se comporta de una forma extraña (al igual que ocurre con otros clasificadores que usan PCA) y las predicciones tienen un error muy alto y en la mayoría de los casos, son estimaciones por debajo de la temperatura real.

Coincide con los límites de temperatura de los modelos DUST. Habría que considerar emplear un mayor numero de componentes en el cálculo de PCA.

A continuación, se presenta un estudio del clasificador PCA+GPS para los conjuntos de validación CONDG202Y y DUSTG202Y, es decir, sobre conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación

La tabla 36, muestra los resultados de reevaluar el clasificador para los conjuntos de espectros de validación CONDG202Yareal y CONDG202Yarealmoving

| | CONDG202Y | CONDG202Y moving |
|---|---|---|
| Correlation coefficient | 0.9573 | 0.956 |
| Mean absolute error | 84.7737 | 86.8824 |
| Root mean squared error | 116.1299 | 117.1491 |
| Relative absolute error | 19.7368% | 20.2277 % |
| Root relative squared error | 22.2363 % | 22.4314 % |

Tabla 36: Resultados para KNN de los conjuntos de espectros CONDG202Yareal

Las figuras 135 y 137 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos CONDG202Yareal y CONDG202Yarealmoving empleando el clasificador PCA−GPS

Las figuras 136 y 138 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDG202Yareal y CONDG202Yarealmoving empleando el clasificador PCA+GPS.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

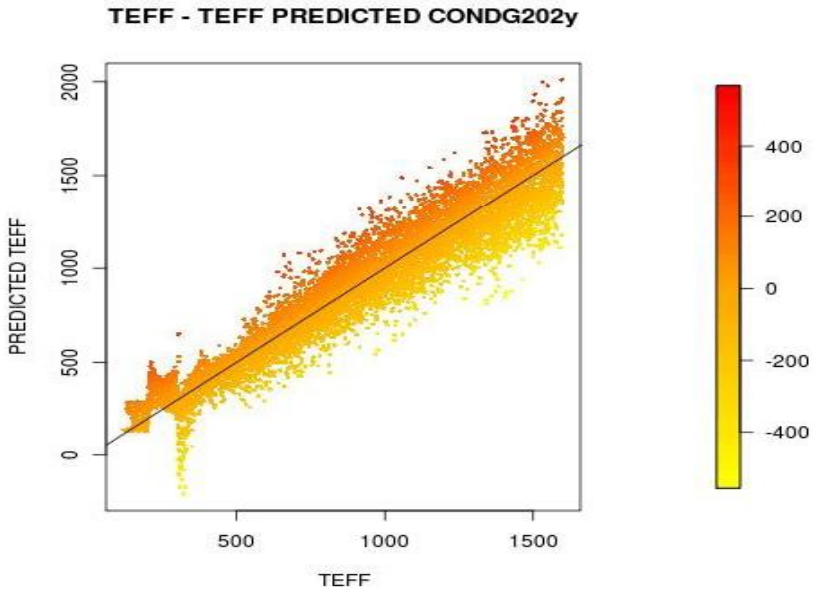


Figura 135 Gratica de dispersión TeIf vs TeIf predicted. sobre PCA−GPS para el conjunto de espectros CONDG202Yarea1.

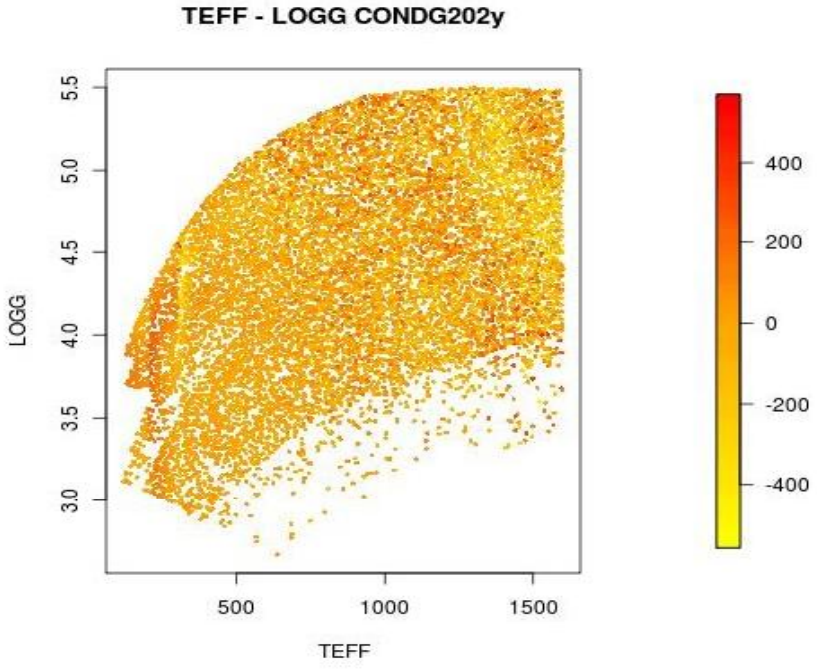


Figura 136. Gratica de dispersión TeIf vs Logg sobre PCA1GPS. para el conjunto de espectros CONDG202Yarea1

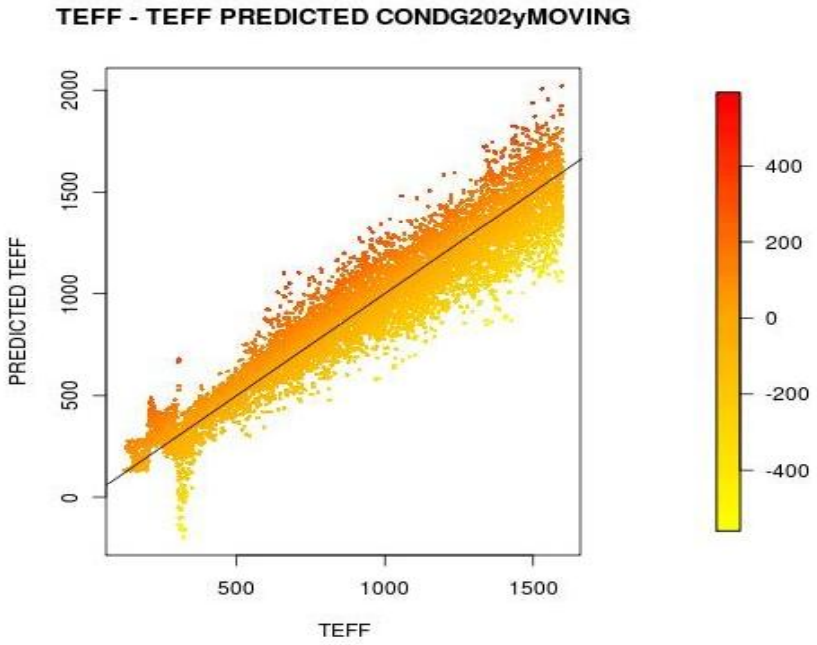


Figura 137 Grafica de dispersión Teff vs Teff predicted, sobre PCA−GPS para el conjunto CONDG202Yarea1moving.

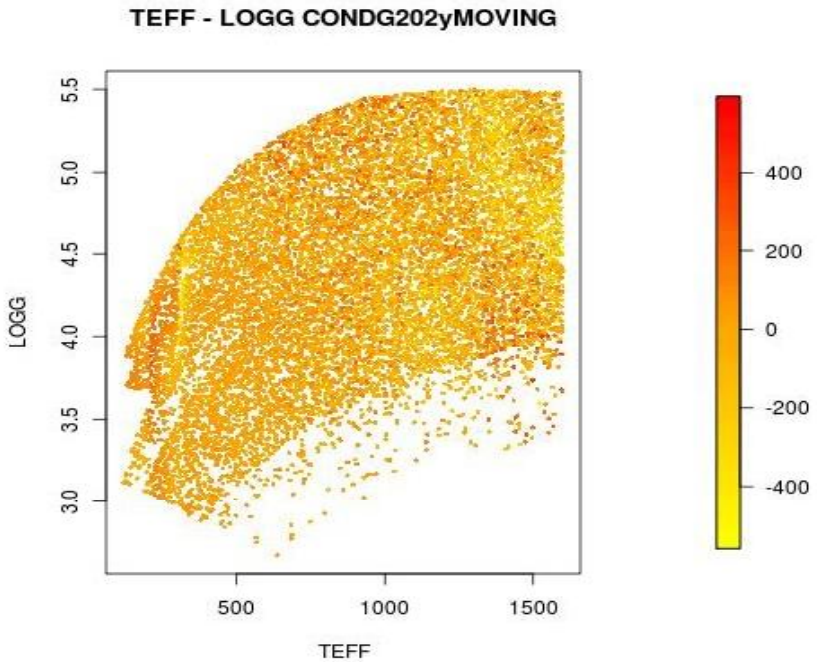


Figura 138. Grafica de dispersión Teff vs Logg sobre PCA−GPS, para el conjunto de espectros CONDG20area1moving

Se observa como para el conjunto de espectros CONDG202Yarea1, la aplicación del suavizado no mejora los resultados

Sigue observándose que existe un error entorno a los 400 ° Kelvin.

Ya no es tan visible que el clasificador realice mejores predicciones en el rango ente los 500 y los 1300° Kelvin, aunque se nota el aumento del error debido al ruido introducido y la falta de información que todavía la Sonda no ha recogido

Los errores de predicción son muy elevados lo que hacen inviable considerar este clasificador para predicción sobre los modelos COND en etapas intermedias de la misión.

A continuación, la tabla 37 nos muestra el resultado de la predicción para los conjuntos de espectros de validación DUSTG202Yarea1 y DUSTG202Yarea1moving

| | DUSTG202Y | DUSTG202Y moving |
|---|---|---|
| Correlation coefficient | 0.8072 | 0.8233 |
| Mean absolute error | 141.6926 | 137.9115 |
| Root mean squared error | 181.2949 | 175.8522 |
| Relative absolute error | 20.8167 % | 20.2612 % |
| Root relative squared error | 24.6647 % | 23.9242 % |

Tabla 37: Resultados para PCA+GPS de los conjuntos de espectros DUSTG202Yarea1, DUSTG202Yarea1moving

Las figuras 139 y 141 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador PCA1GPS.

Las figuras 140 y 142 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos DUSTG202Yarea1 y DUSTG202Yarea1moving empleando el clasificador PCA1GPS.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la

predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

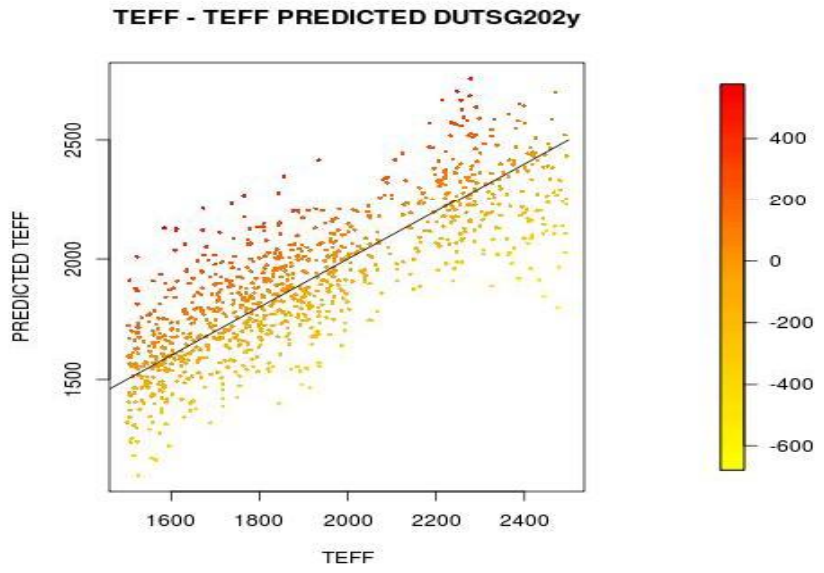


Figura 139 Gráfica de dispersión Teff vs Teff predicted, sobre PCA+GPS para el conjunto de espectros DUSTG202Yarea1.

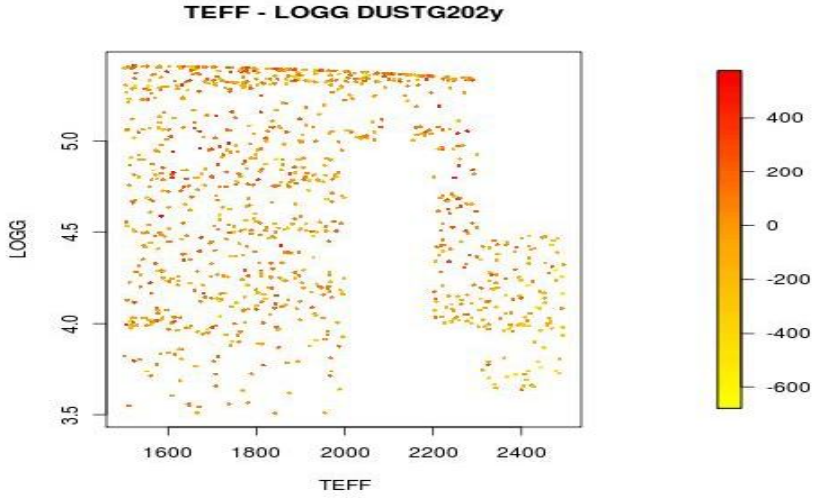


Figura 140. Gráfica de dispersión Teff vs Logg sobre KNN, para el conjunto de espectros DUSTG202Yarea1

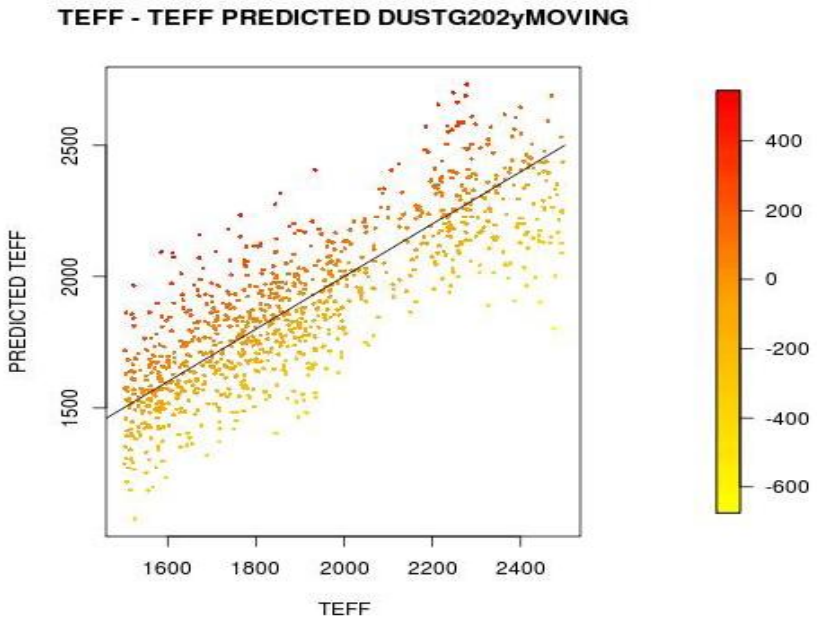


Figura 141 Grafica de dispersión TeIf vs TeIf predicted, sobre PCA−GPS para el conjunto DUSTG20area1moving.

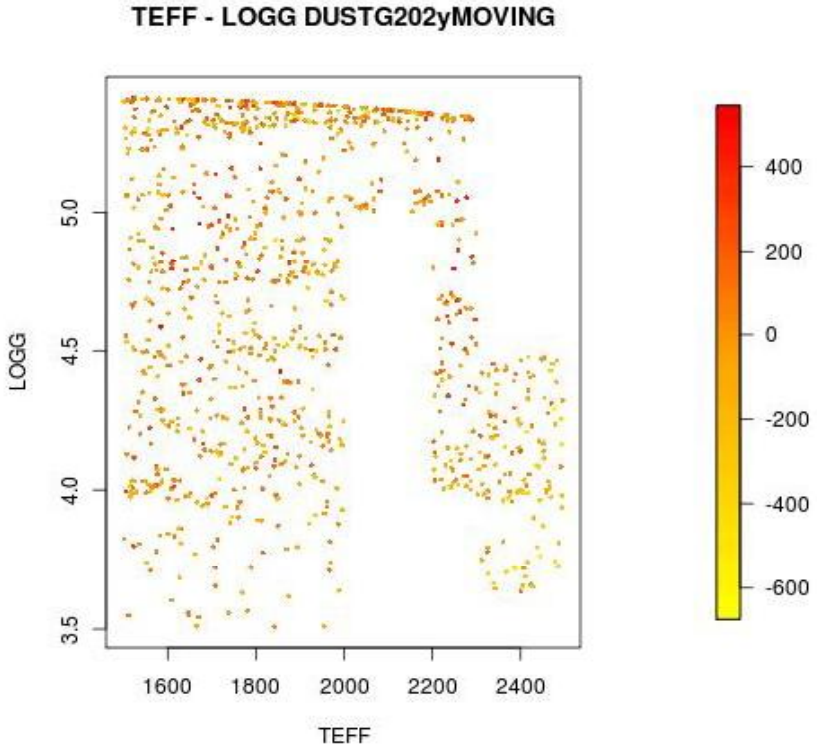


Figura 142 Grafica de dispersión TeIf vs Logg sobre PCA1GPS, para el conjunto de espectros DUSTG20area1moving

Los conjuntos de espectros que pretenden simular los primeros resultados de la sonda GAIA, a poco menos de la mitad de observación, por lo tanto, es normal que los resultados obtenidos sean peores que los conjuntos de datos de validación DUSTG20area1

Para el conjunto de espectros DUSTG202Yarea1, al igual que para el conjunto de espectros estudiados de CONDG202Yarea1 para PCA+GPS, la aplicación del suavizado no mejora los resultados.

El error en temperaturas efectivas superiores a 2200 ° Kelvin coincide en posición con los mismos errores de predicción de los otros clasificadores, lo hace presuponer que existe algún tipo de pérdida de información al aplicar la transformada PCA

Los errores de predicción son muy elevados lo que hacen inviable considerar este clasificador para predicción sobre los modelos DUST en etapas intermedias de la misión

#### 3.2.2.1.4 Conclusiones Parciales

Aplicando la transformada PCA sobre los datos normalizados, aportan una reducción de dimensionalidad, mejorando la predicción empleando KNN sobre los modelos DUST

Sin embargo, como ya se ha repetido en varias ocasiones, se debería estudiar mediante Inferencias Bayesianas o T-Student la bondad de los clasificadores para poder obtener una conclusión real.

Los errores máximos, para todos los clasificadores se concentran entorno a 400 °K, 1600°K y 2200 °K, lo cual nos hace suponer que puedan existir algunos problemas con el número de componentes elegido para la aplicación de la Transformación PCA

Por lo tanto, parece evidente que la decisión de considerar el 95% de la varianza para determinar el número de componentes no ha sido la acertada

Estos comportamientos desvirtuan el funcionamiento general de los clasificadores que, quitando estas excepciones, mantienen un comportamiento predictivo bastante constante para rangos de temperatura entre 600 y 1300 ° Kelvin

Seria interesante, y es motivo de investigación en la tesis Doctoral que da continuación al presente Trabajo Fin de Master, estudiar un mayor número de componentes en la transformada PCA.

#### 3.2.2.2 Resultados para Clasificadores con Preprocesado DiffusionMaps

Otro tipo de transformación y reducción de atributos es mediante la técnica de Mapas de Difusión. Como ya se ha comentado, es una técnica para reducción de dimensionalidad que suele emplearse cuando los espectros del conjunto de datos no se encuentran uniformemente distribuidos. DiffusionMaps incorporan un componente de aleatoriedad totalmente dependiente del conjunto de datos sobre el que se aplica la transformada

Por ese motivo, necesitamos definir una nueva sistemática y estudio de resultados obtenidos con DiffusionMaps para los clasificadores KNN, SMO y Gps.

La problemática del estudio de los DiffusionMaps viene determinada por su aleatoriedad relacionada con el conjunto de datos como hemos comentado. El sistema introduce cambios de signo aleatorio sobre las nuevas componentes

Esto nos obliga a tener que aplicar la transformada de los Mapas de Difusión sobre el conjunto de entrenamiento y el conjunto de validación simultaneamente

Por lo tanto, tendremos que generar conjuntos de espectros (entrenamiento – validación) únicos para cada experimento

Además, existe otro problema añadido, un problema de limitación de número de espectros sobre el paquete estadístico R. No podemos aplicar una transformada de DiffusionMaps sobre un conjunto

de datos superior a 4500 espectros

Los conjuntos de espectros de validación CONDRAN y CONDG20 constan de 10.000 espectros.

El conjunto de espectros de validación DUST sólo tiene 1.000 espectros.

Esto nos obliga a tener que subdividir los conjuntos de datos CONDG20 y RANCOND. El número de pruebas y experimentos se multiplica

Para evaluar el sistema de las DifusionMaps, se va definir un conjunto experimentos menos profundo que al menos nos permita comparar con los clasificadores anteriores

Se decide evaluar los clasificadores KNN, SMO y Gps partiendo de los conjuntos de datos NOMarea1 para validación y únicamente RANCONDG15area1, RANDUSTG15area1, CONDG20area1 y DUSTG20area1 para validar

Se define la siguiente metodología de obtención de conjuntos de datos a aplicar Mapas de Difusión:

1) Se dividen los conjuntos de datos RANG15area1 y CONDG20area1 en tres subconjuntos cada uno, etiquetados como RAN1, RAN2, RAN3, G201, G202 Y G203, con el siguiente numero de instancias respectivamente: 4.000, 4,000, 4,000, 3,500, 3,500 y 3.000

2) Se agrupan los conjuntos de espectros de validación RANDUSTG15area1 y DUSTG20area1 en un único conjunto de datos llamado DUST.

3) A los subconjuntos de datos obtenidos: RANCOND1, RANCOND2, RANCOND3, CONDG201, CONDG202, CONDG203 y DUSTG20, se les suma el conjunto de datos de entrenamiento NOMarea1 consistente en 564 espectros, quedándose determinados los conjuntos de aplicación de los Mapas de Difusión: NOMRANCOND1, NOMRANCOND2, NOMRANCOND3, NOMCONDG201, NOMCONDG202, NOMCONDG203 y

NOMDUSTG20 presentados en la tabla 38

| CONJUNTO DE DATOS | Nº INSTANCIAS |
|---|---|
| NOMRANCOND1 | 4164 |
| NOMRANCOND2 | 4164 |
| NOMRANCOND3 | 3664 |
| NOMCONDG201 | 4564 |
| NOMCONDG202 | 4564 |
| NOMCONDG203 | 3664 |
| NOMDUSTG20 | 2564 |

Tabla 38 Conjunto de datos a transformar por DiffusionMaps en R

La transformada DiffusionMaps depende de dos variables epsilon.value y neigen

- El parámetro epsion, para el peso de difusión de la matriz de pesos: exp(-D$^2$/(eps.val)). Por defecto epsilon usa la distancia media a n* 0,01 vecinos más cercanos.

- Neigen determina el número de dimensiones de la representación final del mapa de difusión. Por defecto utiliza un número de dimensiones correspondientes a un 95% de descenso del valor del propio multiplicador.

Se realiza un estudio combinatorio de diferentes valores de dimensionalidad y epsilon empleando para ello KNN como clasificador y la técnica de validación cruzada sobre el conjunto de datos de entrenamiento NOMarca1.

El estudio nos determina que los valores idóneos de trabajo son

- Epsilon – 0.5

- Neigen – 7

4) Una vez aplicada la transformada sobre los conjuntos de datos definidos en la tabla 39, se vuelven a desagrupar los datos, obteniendo las parejas de aplicación de los Clasificadores en Weka

| Conjunto de entrenamiento | Conjunto de Validacion |
|---|---|
| NOMCONDDM1 | RANCONDDM1 |
| NOMCONDDM2 | RANCONDDM2 |
| NOMCONDDM3 | RANCONDDM3 |
| NOMG20DM1 | G20DM1 |
| NOMG20DM2 | G20DM2 |
| NOMG20DM3 | G20DM3 |
| NOMDUSTDM | DUSTDM |
| NOMDUSTDM | DUSTG20DM |

Tabla 39 Parejas de conjuntos de datos a usar en los clasificadores

#### 3.2.2.2.1 Resultados de Mapas de difusion + KNN

El algoritmo de k-vecinos cercanos se implementa en Weka a través del clasificador

weka.classifiers.lazy.IBk

En las mejores condiciones obtenidas, las variables del clasificador quedaron definidas de la siguiente forma:

KNN=5, es decir, empleando los 5 vecinos cercanos

DistanceWeighting = "by 1/distance", es decir, usando una ponderación inversa a la distancia de cada vecino cercano.

- NearestNeighbourSearchAlgorithm = "CoverTree", La busqueda del vecino cercano emplea el algoritmo "covertree" que considera una estructura de arbol con jerarquia de niveles, conteniendo todos los puntos del espacio métrico

Los resultados obtenidos de los modelos de COND para los clasificadores (una vez optimizados las

variables de cada uno de ellos) se pueden ver detallados en la Tabla 40

| NOMCONDDM1 | NOMCONDDM2 | NOMCONDDM1 |
|---|---|---|
| 0.9997 | 0.9997 | 0,9999 |
| 7.928 | 7.8367 | 1,0638 |
| 15.7126 | 15.3709 | 10,3142 |
| 1.3744 % | 1.3585 % | 0,18% |
| 2.3139 % | 2.2636 % | 1,52% |
| RANCONDDM1 | RANCONDDM2 | RANCONDDM3 |
| 0.9896 | 0.9865 | 0.9881 |
| 34.8307 | 35.8434 | 35.181 |
| 61.7563 | 69.9118 | 65.7952 |
| 7.8899 % | 8.5512 % | 8.2261 % |
| 11.6184 % | 13.6166 % | 12.6156 % |
| G20DM1 | G20DM2 | G20DM3 |
| 0,9855 | 0,9815 | 0,9823 |
| 43,6502 | 44,9324 | 45,2132 |
| 72,2465 | 81,0013 | 79,9026 |
| 9,89% | 10,72% | 10,57% |
| 13,59% | 15,78% | 15,32% |

Tabla 40 Resultados obtenidos con DiffusionMaps KNN para los conjuntos de datos RANarea1 y CONDG20area1

Para la representación gráfica se han unido los resultados predictivos de RANCONDDM1, RANCONDDM2 y RANCONDDM3, para mostrar una única representación gráfica en las figuras 143 y 144

Para la representación gráfica se han unido los resultados predictivos de G20DM1, G20DM2 y G20DM3, para mostrar una única representación gráfica en las figuras 145 y 146.

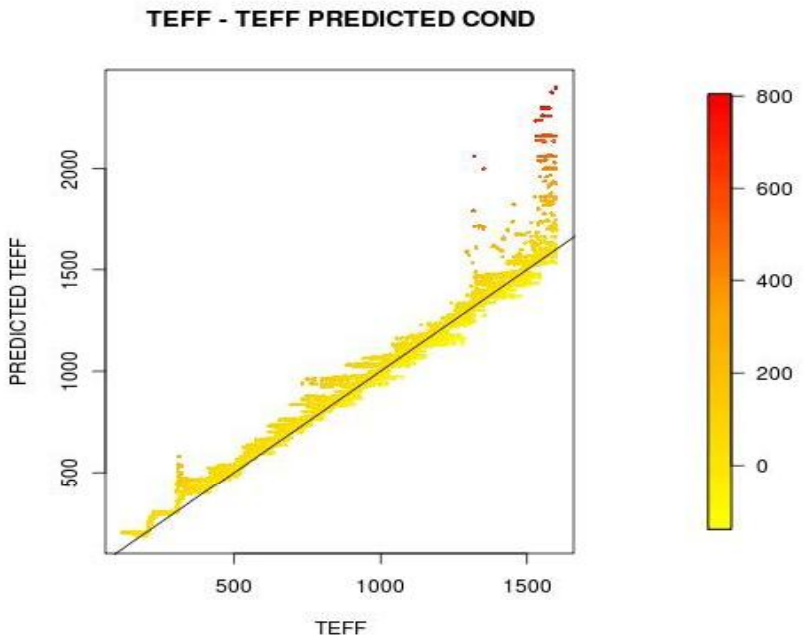


Figura 143 Gráfica de dispersión TEFF TEFF PREDICTED para la predicción sobre DiffusionMaps KNN de TEFF para los conjuntos de validación CONDRANDM1, 2 y 3 .

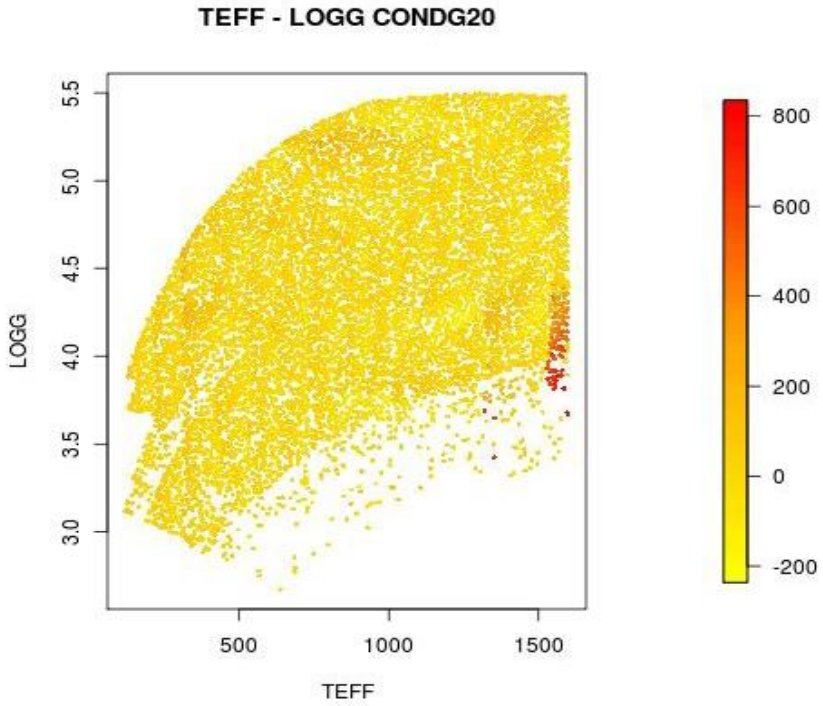


Figura 144 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps KNN de TEFF para el conjunto de validación CONDRANDM1, 2 y 3

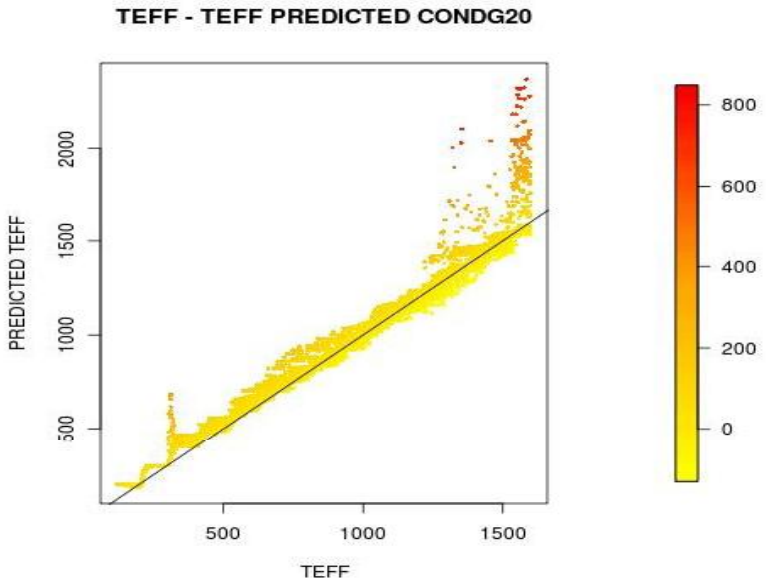


Figura 145 Gráfica de dispersión TEFF TEFF PREDICTED para la predicción sobre DiffusionMaps KNN de TEFF para el conjunto de validación G20DM1, 2 y 3.

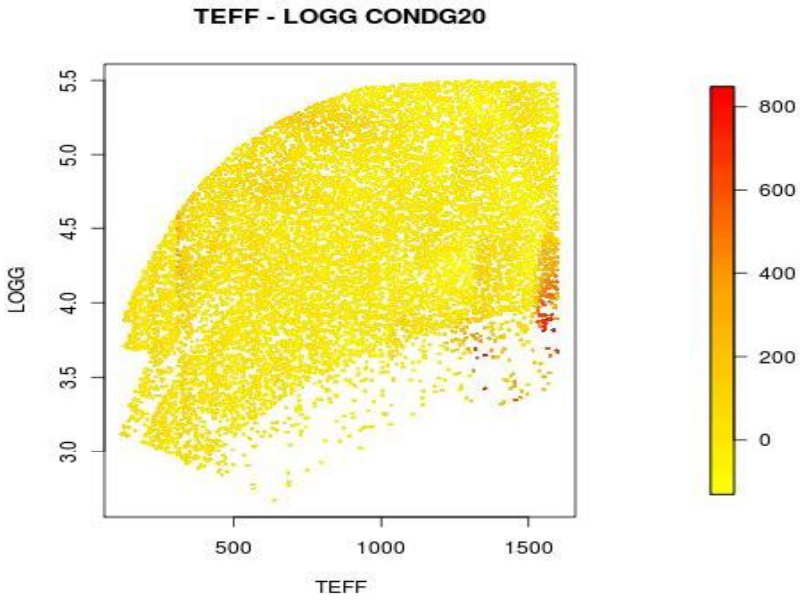


Figura 146 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps-KNN de TEFF para el conjunto de validación G20DM1, 2 y 3.

Se puede observar como el comportamiento de los Mapas de Difusión es muy bueno para temperaturas entre 500° y 1300° K.

Sin embargo, los errores se producen por encima de 1500° pero son errores muy elevados, valores incluso por encima de 700°, que desvirtuan las precisiones conseguidas en Temperatura efectiva reales.

Este error parece que se encuentra marcado en los límites de temperatura del modelo COND, sería interesante analizar este comportamiento en esta zona concreta.

Pasa lo mismo que con PCA, en el rango final de temperaturas, el clasificador genera unos errores

Por debajo de 500° K, el clasificador forma un escalado que podría coincidir con el ya comentado error de interpolación

En temperaturas reales de 400 °K existe un error en la predicción. Este error ya se ha presentado con anterioridad en los clasificadores con PCA, sería interesante profundizar en el porqué

Los resultados obtenidos de los modelos de DUST para los clasificadores (una vez optimizados las variables de cada uno de ellos) se pueden ver detallados en la Tabla 41, en este caso, la transformada de los Mapas de Difusión se realizó una única vez sobre un mismo conjunto de datos que incluía el conjunto de entrenamiento NOMDUSTDM y los conjuntos de validación DUSTDM y DUSTG20DM.

| | DUSTDM | DUSTG20 |
|---|---|---|
| Correlation coefficient | 0.9922 | 0.9944 |
| Mean absolute error | 25.5276 | 23.9238 |
| Root mean squared error | 36.8379 | 32.9005 |
| Relative absolute error | 3.7504 % | 3.5148 % |
| Root relative squared error | 5.0117 % | 4.476 % |

Tabla 41 Resultados obtenidos con DiffusionMaps – KNN para los conjuntos de datos DUSTDM y DUSTG20

Las figuras 147 y 149 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación

DUSTDM y DUSTG20 empleando el clasificador DiffusionMaps+KNN

Las figuras 148 y 150 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos DUSTDM y DUSTG20 empleando el clasificador DiffusionMaps+KNN.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

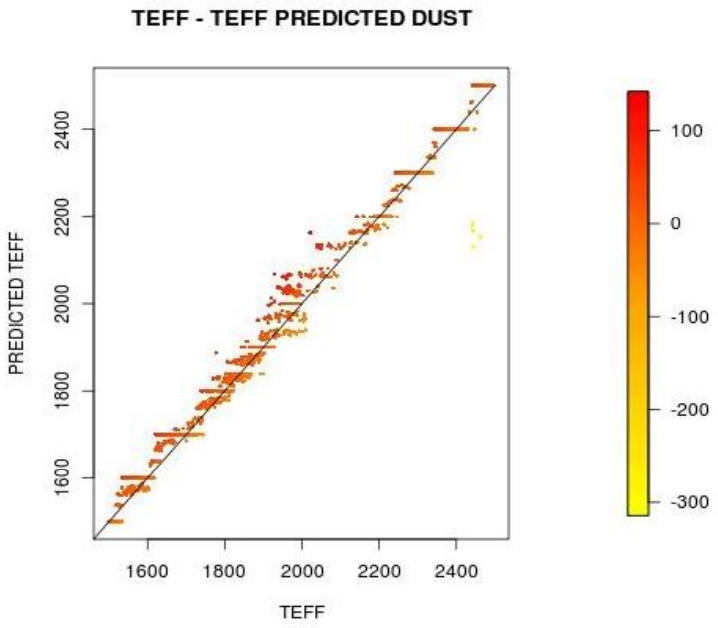


Figura 147 Gráfica de dispersión TEFF - TEFF PREDICTED para la predicción sobre DiffusionMaps+KNN de TEFF para el conjunto de validación DUSTDM.

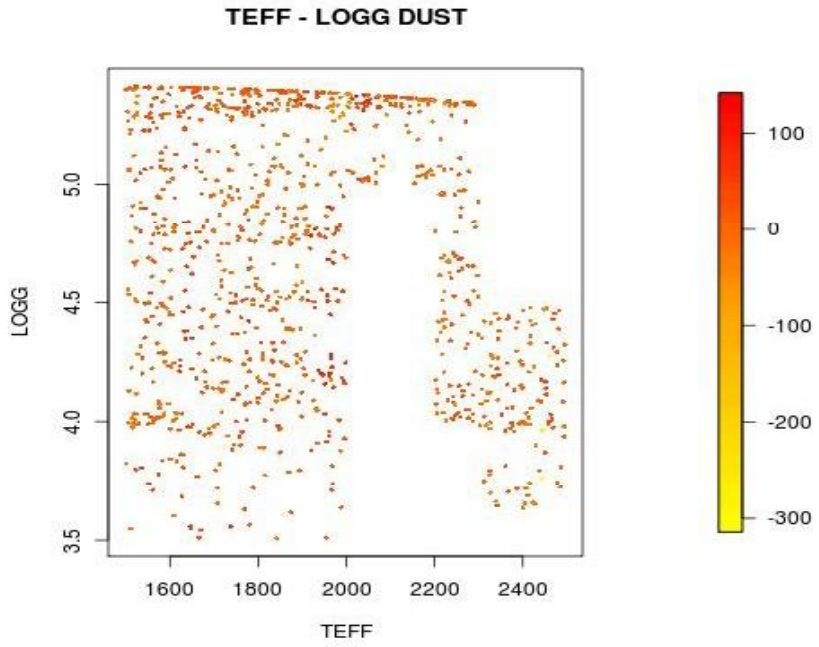


Figura 148 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps+KNN de TEFF para el conjunto de validación DUSTDM.

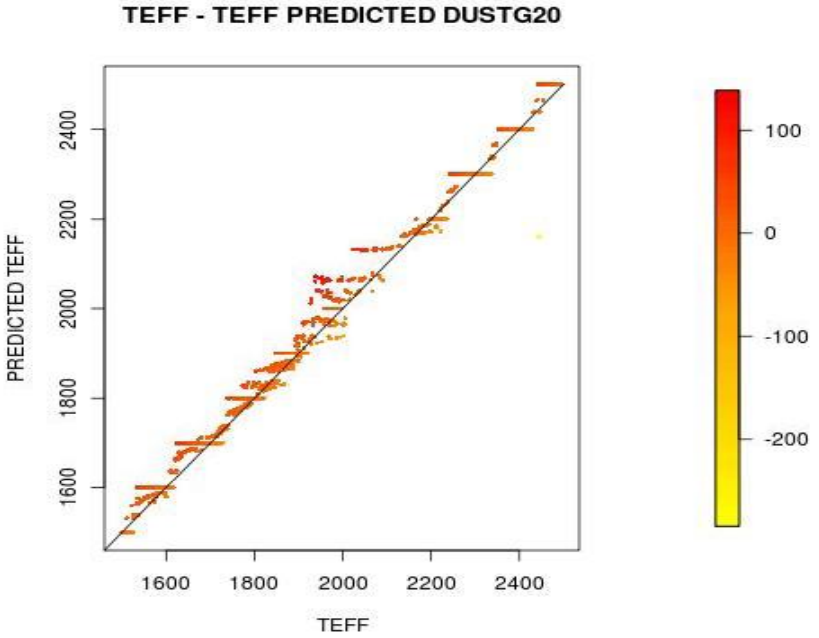


Figura 149 Gráfica de dispersión TEFF TEFF PREDICTED para la predicción sobre DiffusionMaps+KNN de TEFF para el conjunto de validación DUSTG20DM.

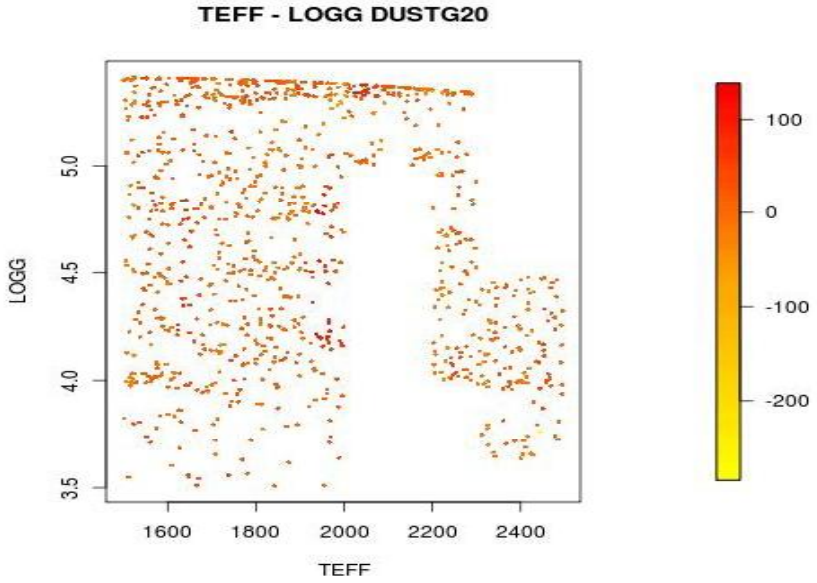


Figura 150 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps+KNN de TEFF para el conjunto de validación DUSTG20DM.

Se puede observar como el comportamiento de los diffusionMaps con el clasificador KNN es muy regular para todo el rango de temperaturas de los modelos DUST salvo para un caso puntual que habría que estudiar más al detalle.

Existe un error de predicción focalizado en los 2400 ° K que afecta a la predicción de temperatura en el modelo DUST. Habría que indagar el motivo por el cual aparece esta predicción tan mál clasificada

En la tesis doctoral que dá continuidad al presente Trabajo Fin de Master, se profundizará en la investigación de este tipo de clasificador ya que su gran regularidad es una de sus principales virtudes.

#### 3.2.2.2.2 Resultados para Mapas de difusión + SMO

Partiendo del mismo procedimiento de experimentación comentado en 3.2.2.2.1, se han realizado diferentes experimentaciones para obtener el mejor sistema de predicción sobre máquinas de vectores soporte, usando los Mapas de Difusión como reductores de dimensionalidad

A continuación, en la tabla 42, se muestran los resultados obtenidos empleando los datos transformados por mapas de difusión en máquinas de vectores soporte para el rango de modelos COND.

Se han optimizado el factor gamma del kernel (g) al valor 18 y el margen blando (c) de la máquina de vectores al valor a 1,2.

| | NOMCONDDM1 | NOMCONDDM2 | NOMCONDDM3 |
|---|---|---|---|
| Correlation coefficient | 0,9988 | 0,9989 | 0,9991 |
| Mean absolute error | 5,0904 | 8,6727 | 8,0383 |
| Root mean squared error | 34,7626 | 32,9921 | 29,4457 |
| Relative absolute error | 1,58% | 1,50% | 1,39% |
| Root relative squared error | 5,12% | 4,86% | 4,34% |
| | RANCONDDM1 | RANCONDDM2 | RANCONDDM3 |
| Correlation coefficient | 0,9957 | 0,9942 | 0,9949 |
| Mean absolute error | 31,7397 | 33,6385 | 32,0109 |
| Root mean squared error | 46,0117 | 52,0503 | 49,4531 |
| Relative absolute error | 7,19% | 8,03% | 7,48% |
| Root relative squared error | 8,66% | 10,14% | 9,48% |
| | G20DM1 | G20DM2 | G20DM3 |
| Correlation coefficient | 0,9916 | 0,9898 | 0,9905 |
| Mean absolute error | 33,522 | 36,2535 | 34,5296 |
| Root mean squared error | 58,8585 | 65,2052 | 62,2567 |
| Relative absolute error | 7,62% | 8,65% | 8,07% |
| Root relative squared error | 11,07% | 12,72% | 11,94% |

Tabla 42 Resultados obtenidos con DiffusionMaps – SMO para los conjuntos de datos CONDDM y CONDG20

Para la representación gráfica se han unido los resultados predictivos de RANCONDDM1, RANCONDDM2 y RANCONDDM3, para mostrar una única representación gráfica en las figuras 151 y 152

Para la representación gráfica se han unido los resultados predictivos de G20DM1, G20DM2 y G20DM3, para mostrar una única representación gráfica en las figuras 153 y 154.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

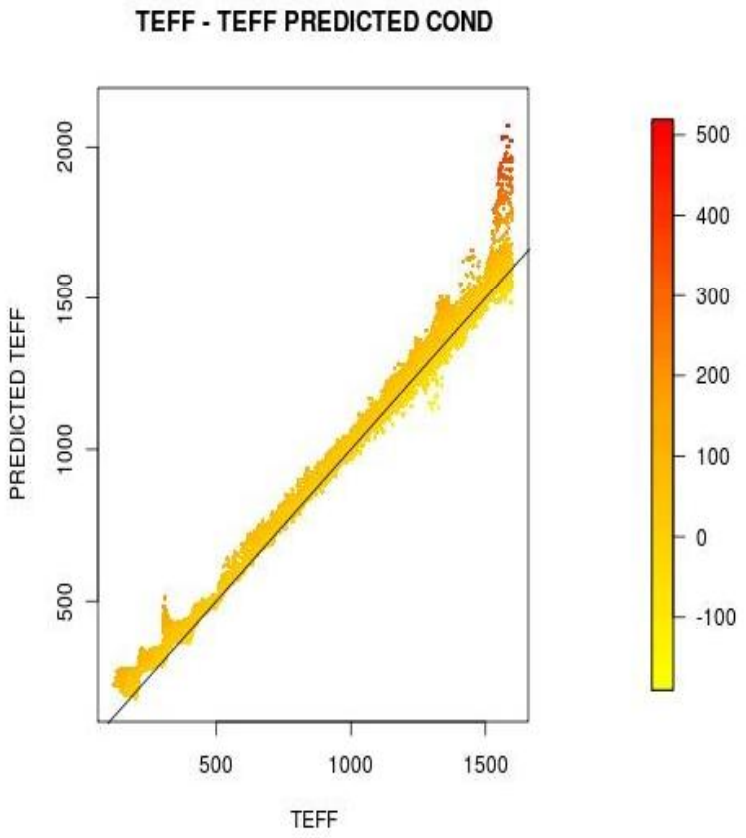


Figura 151 Gráfica de dispersión TEFF – TEFF PREDICTED para la predicción sobre DiffusionMaps+SMO de TEFF para el conjunto de validación COND_DM.

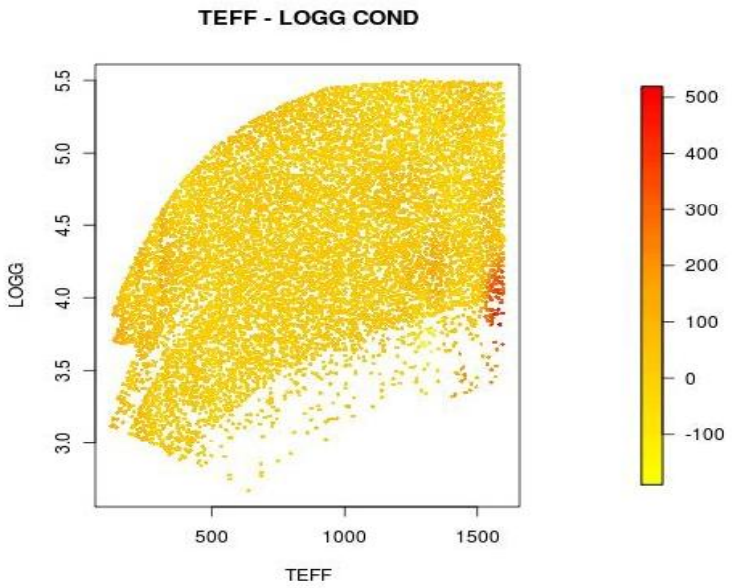


Figura 152 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps SMO de TEFF para el conjunto de validación CONDDM.

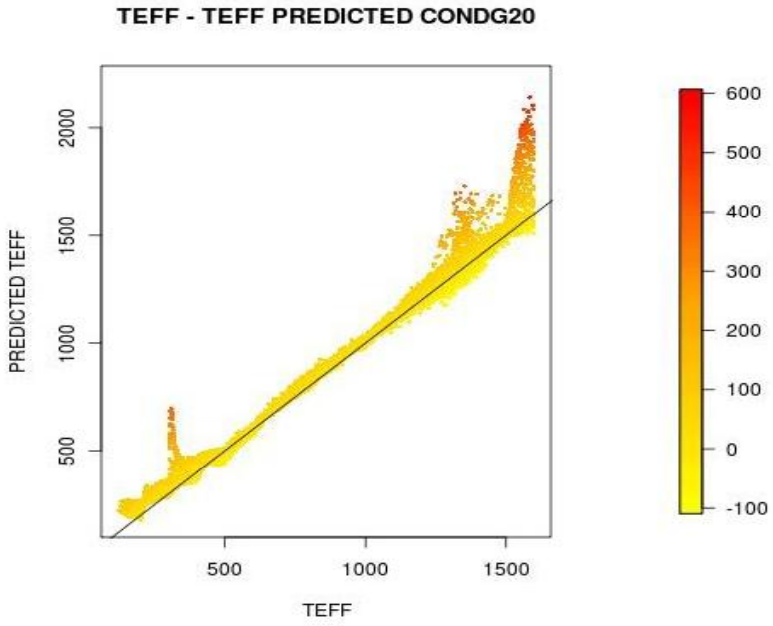


Figura 153 Gráfica de dispersión TEFF TEFF PREDICTED para la predicción sobre DiffusionMaps SMO de TEFF para el conjunto de validación CONDG20DM.

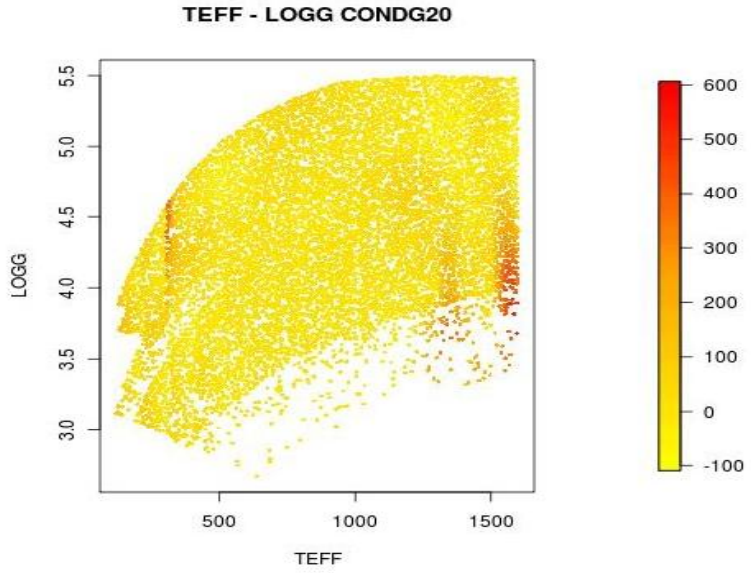


Figura 154 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps+KNN de TEFF para el conjunto de validación CONDG20DM.

El mismo efecto y problemática nos encontramos con los K vecinos cercanos y máquinas de vectores:

Se puede observar como el comportamiento es muy bueno para temperaturas entre 500° y 1300° K

Sin embargo, los errores se producen por encima de 1300° pero son errores muy elevados que desvirtuan las predicciones conseguidas.

Este error parece que se encuentra marcado en los límites de temperatura del modelo COND, sería interesante analizar este comportamiento en esta zona concreta

Pasa lo mismo que con PCA, en el rango final de temperaturas, el clasificador genera unos errores.

Por debajo de 500° K, el clasificador forma un escalado que podría coincidir con el ya comentado error de interpolación.

En temperaturas reales de 400 °K existe un error en la predicción. Este error ya se ha presentado con anterioridad en los clasificadores con PCA y Mapas de Difusión, sería interesante profundizar en el

motivo

La tabla 43 presenta los resultados obtenidos para los modelos de DUST, en los conjuntos de datos para validación DUSTDM y DUSTG20.

| | DUSTDM | DUSTG20 |
|---|---|---|
| Correlation coefficient | 0.9761 | 0.9811 |
| Mean absolute error | 44.8069 | 40.5929 |
| Root mean squared error | 67.554 | 62.2776 |
| Relative absolute error | 6.5828 % | 5.9637 % |
| Root relative squared error | 9.1905 % | 8.4727 % |

Tabla 43 Resultados obtenidos con DiffusionMaps SMO para los conjuntos de datos DUSTDM y DUSTG20

Las figuras 155 y 157 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación DUSTDM y DUSTG20 empleando el clasificador DiffusionMaps+SMO

Las figuras 156 y 158 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos DUSTDM y DUSTG20 empleando el clasificador DiffusionMaps+SMO.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

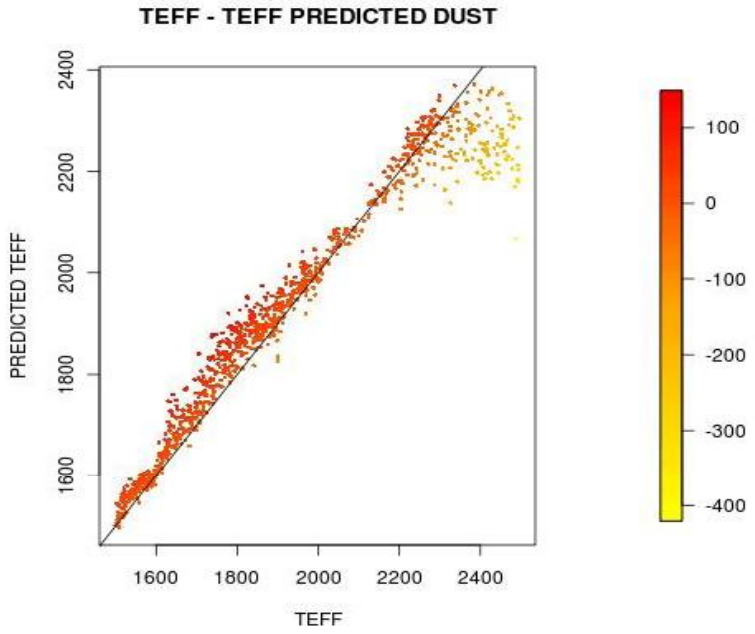


Figura 155 Gráfica de dispersión TEFF TEFF PREDICTED para la predicción sobre DiffusionMaps SMO de TEFF para el conjunto de validación DUSTDM.

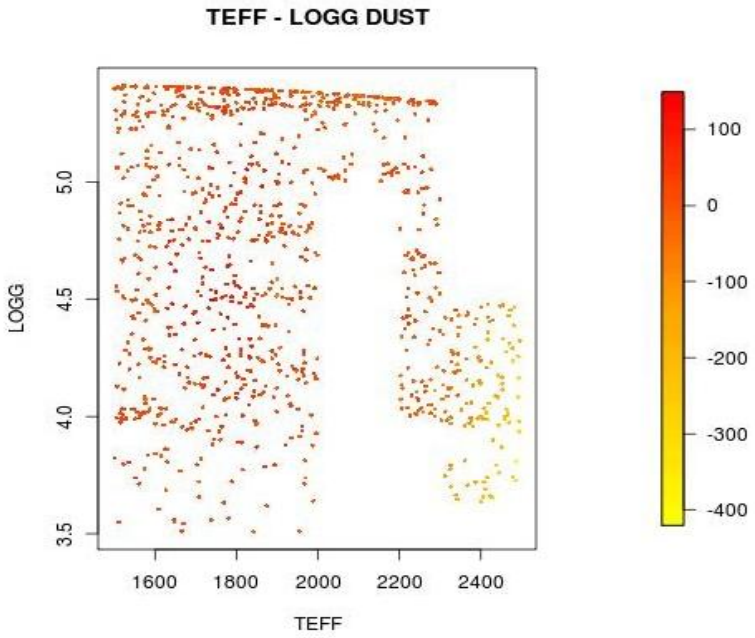


Figura 156 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps SMO de TEFF para el conjunto de validación DUSTDM.

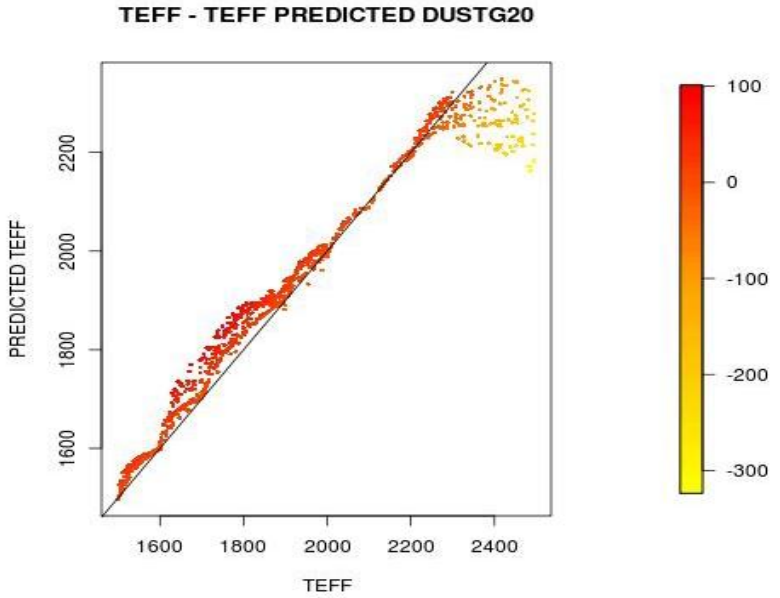


Figura 157 Gráfica de dispersión TEFF TEFF PREDICTED para la predicción sobre DiffusionMaps+SMO de TEFF para el conjunto de validación DUSTG20DM.

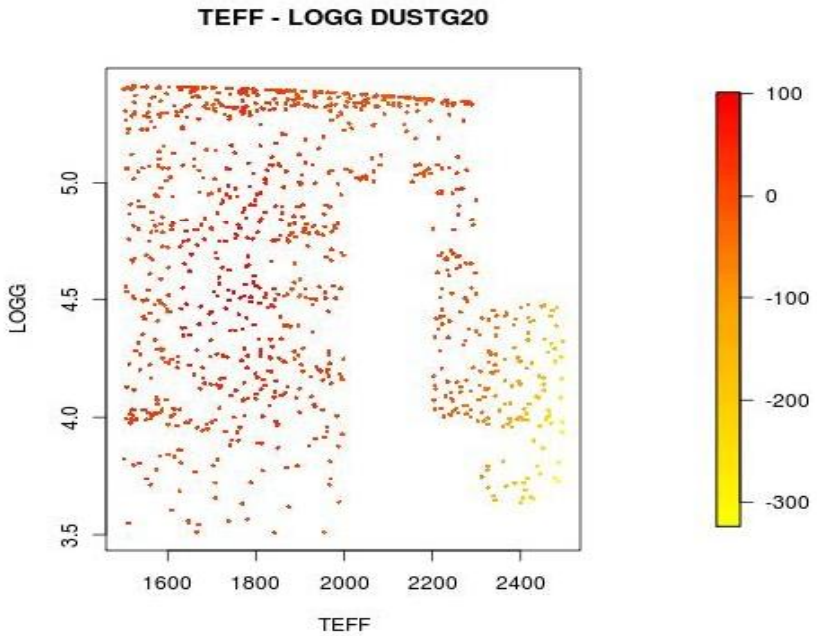


Figura 158 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps+KNN de TEFF para el conjunto de validación DUSTG20DM.

Se puede observar como el comportamiento de los diffusionMaps con el clasificador SMO es muy regular para todo el rango de temperaturas de los modelos DUST salvo para un caso puntual que habria que estudiar más al detalle.

Existe un error de predicción focalizado en los 2400 ° K que afecta a la predicción de temperatura en el modelo DUST. Habria que indagar el motivo por el cual aparece esta predicción tan mál clasificada

A priori parece que existe un problema con las predicciones en el rango final de las temperaturas del modelo DUST. Este error ha aparecido constantemente en otros clasificadores. Habria que estudiar si es motivos de reducción de dimensionalidad.

#### 3.2.2.2.3 Resultados de Mapas de Difusión más Procesos Gausianos

Partiendo del mismo procedimiento de experimentación comentado en 3.2.2.2.1, se han realizado diferentes experimentaciones para obtener el mejor sistema de predicción sobre máquinas de vectores soporte, usando los Mapas de Difusión como reductores de dimensionalidad

A continuación, en la tabla 44, se muestran los resultados obtenidos empleando los datos transformados por mapas de difusión en procesos gausianos para el rango de modelos COND.

Se han optimizado el factor gamma del kernel (g) al valor 18 y el margen blando (c) de la máquina de vectores al valor a 1,2.

| | NOMCONDDM1 | NOMCONDDM2 | NOMCONDDM3 |
|---|---|---|---|
| Correlation coefficient | 0,9986 | 0,9984 | 0,9983 |
| Mean absolute error | 31,2183 | 16,4489 | 17,8011 |
| Root mean squared error | 49,1589 | 38,4552 | 40,1236 |
| Relative absolute error | 7,07% | 2,85% | 3,09% |
| Root relative squared error | 9,25% | 5,66% | 5,91% |
| | RANCONDDM1 | RANCONDDM2 | RANCONDDM3 |
| Correlation coefficient | 0,9961 | 0,9947 | 0,9947 |
| Mean absolute error | 27,6756 | 30,8085 | 30,421 |
| Root mean squared error | 30,5903 | 45,8346 | 45,4075 |
| Relative absolute error | 6,27% | 7,35% | 7,11% |
| Root relative squared error | 7,45% | 8,93% | 8,71% |
| | G2DDM1 | G2DDM2 | G2DDM3 |
| Correlation coefficient | 0,9854 | 0,9814 | 0,983 |
| Mean absolute error | 41,8267 | 45,5731 | 43,9618 |
| Root mean squared error | 71,7624 | 81,9714 | 77,3185 |
| Relative absolute error | 9,47% | 10,87% | 10,28% |
| Root relative squared error | 13,50% | 15,97% | 14,83% |

Tabla 44 Resultados obtenidos con DiffusionMaps – GPs para los conjuntos de datos CONDDM y CONDG2D

Las figuras 159 y 161 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación CONDDM y CONDDM2D empleando el clasificador DiffusionMaps+GPs.

Las figuras 160 y 162 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos CONDDM y CONDDM2D empleando el clasificador DiffusionMaps+GPs

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

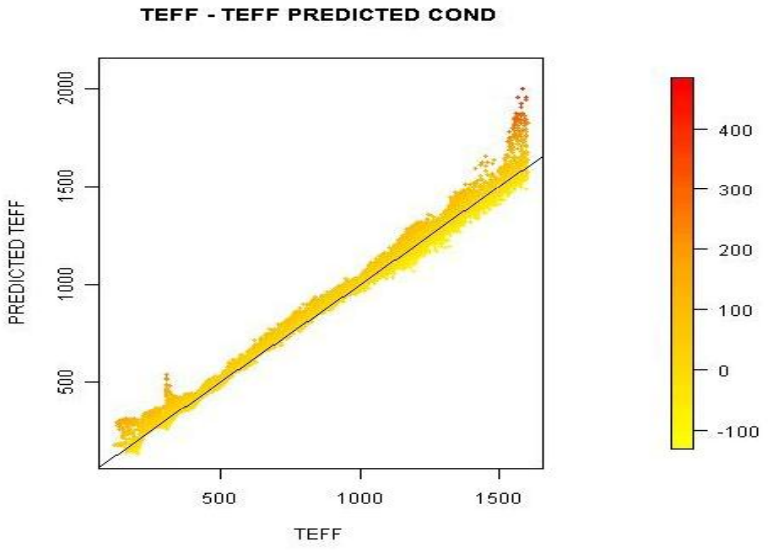


Figura 159 Gráfica de dispersión TEFF TEFF PREDICTED para la predicción sobre DiffusionMaps+GPs de TEFF para el conjunto de validación CONDDM.

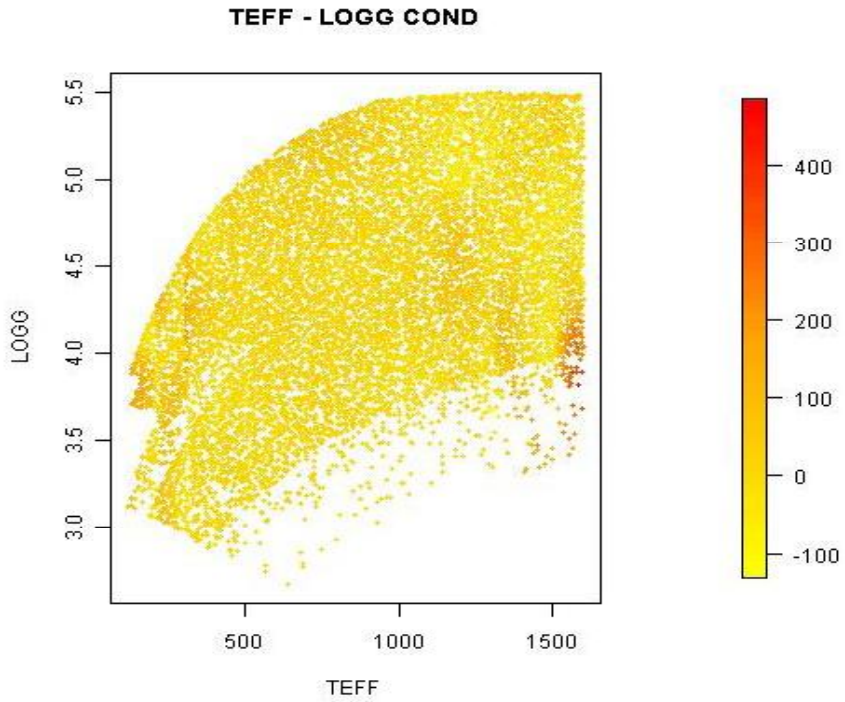


Figura 160 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps+GPs de TEFF para el conjunto de validación CONDDM.

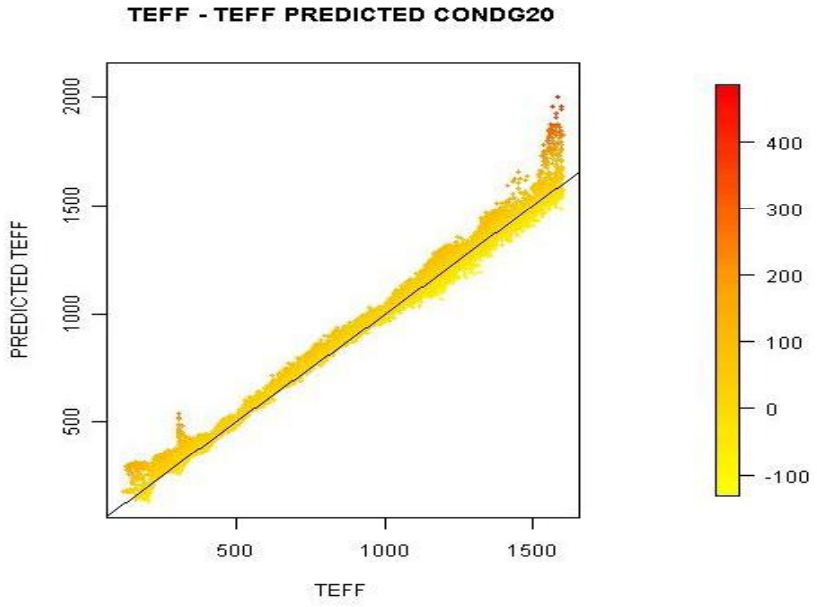


Figura 161 Grafica de dispersión TEFF TEFF PREDICTED para la predicción sobre DiffusionMaps+GPs de TEFF para el conjunto de validación CONDG20DM.

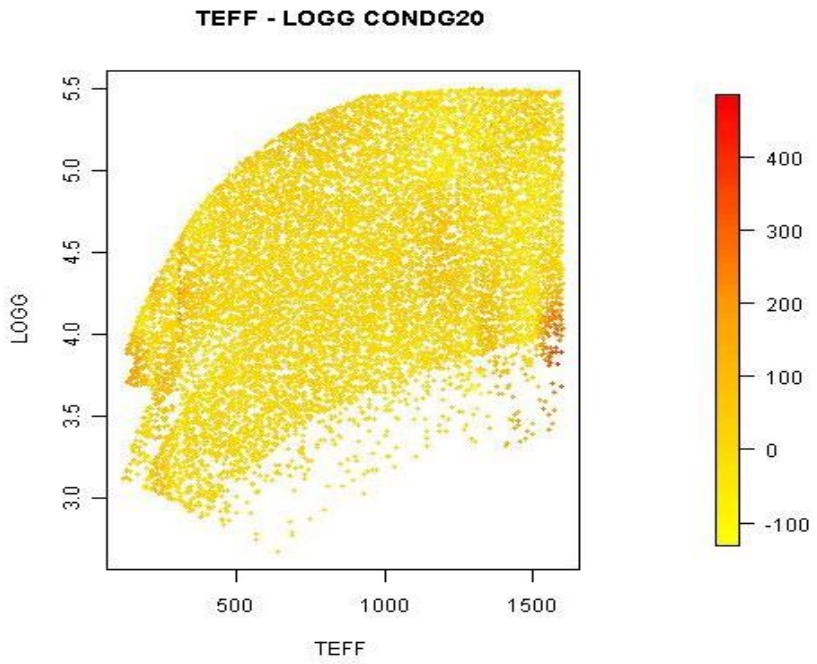


Figura 162 Grafica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps+GPs de TEFF para el conjunto de validación CONDG20DM.

El mismo efecto y problemática nos encontramos con los mapas de difusión y máquinas de vectores. el sistema es muy bueno y realiza predicciones muy buenas por debajo de 1500 grados.

Existe un error en los límites del conjunto de datos de validación COND. este error ya se ha visto en otros clasificadores

En este caso concreto, existe un error para la temperatura efectiva 1500 "K y un logaritmo de la gravedad entorno a 4

La tabla 45 presenta los resultados obtenidos para los modelos de DUST, en los conjuntos de datos para validación DUSTDM y DUSTG20.

| | DUSTDM | DUSTG20 |
|---|---|---|
| Correlation coefficient | 0.951 | 0.9593 |
| Mean absolute error | 70.7035 | 65.8751 |
| Root mean squared error | 94.2802 | 88.8401 |
| Relative absolute error | 10.3874 % | 9.678 % |
| Root relative squared error | 12.8266 % | 12.0865 % |

Tabla 45 Resultados obtenidos con DiffusionMaps + GPs para los conjuntos de datos DUSTDM y DUSTG20

Las figuras 163 y 165 nos muestran respectivamente gráficas de dispersión de temperatura estimada frente a temperatura real para los valores de predicción sobre el conjunto de datos de validación DUSTDM y DUSTDM20 empleando el clasificador DiffusionMaps GPs.

Las figuras 164 y 166 nos muestran respectivamente gráficas de dispersión de temperatura real frente al logaritmo de la gravedad para los valores de predicción sobre el conjunto de datos DUSTDM y DUSTDM20 empleando el clasificador DiffusionMaps+GPs

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas

que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

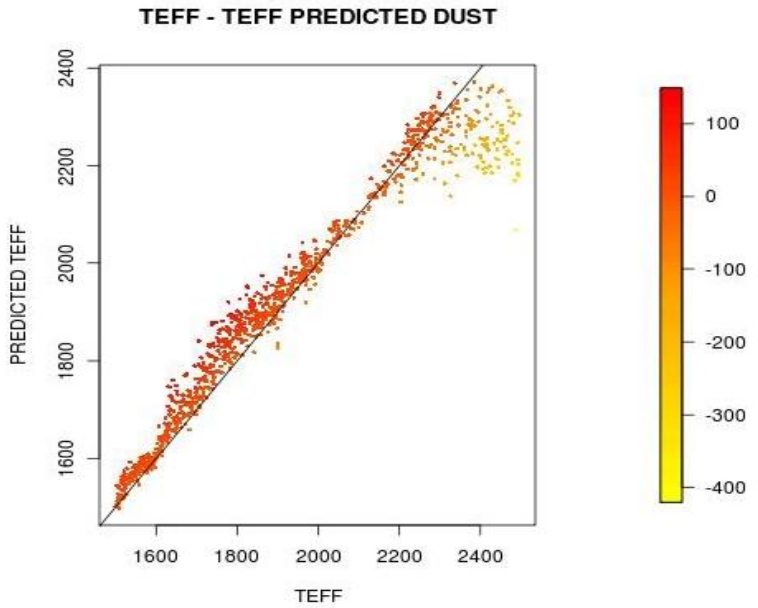


Figura 163 Gráfica de dispersión TEFF TEFF PREDICTED para la predicción sobre DiffusionMaps GPs de TEFF para el conjunto de validación DUSTDM.

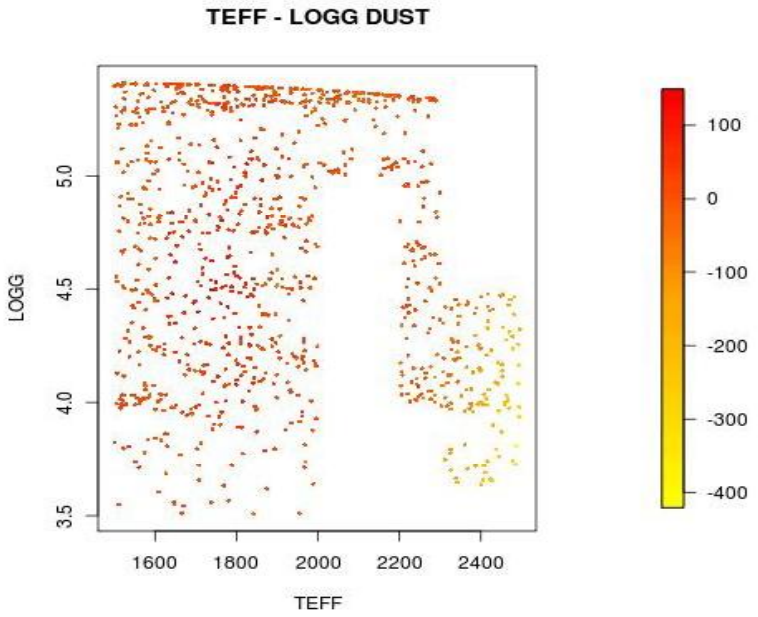


Figura 164 Gráfica de dispersión TEFF LOGG para la predicción sobre DiffusionMaps GPs de TEFF para el conjunto de validación DUSTDM.

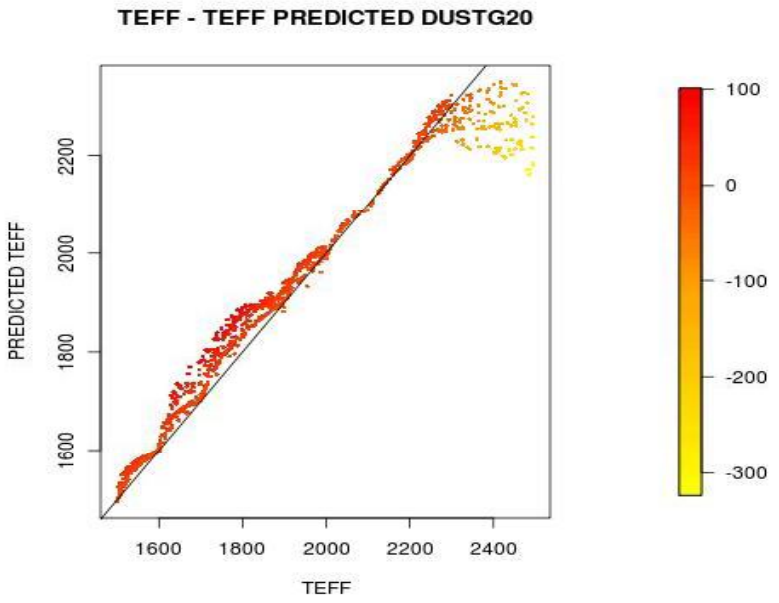


Figura 165 Gráfica de dispersión TEFF – TEFF PREDICTED para la predicción sobre DiffusionMaps GPs de TEFF para el conjunto de validación DUSTG20DM.

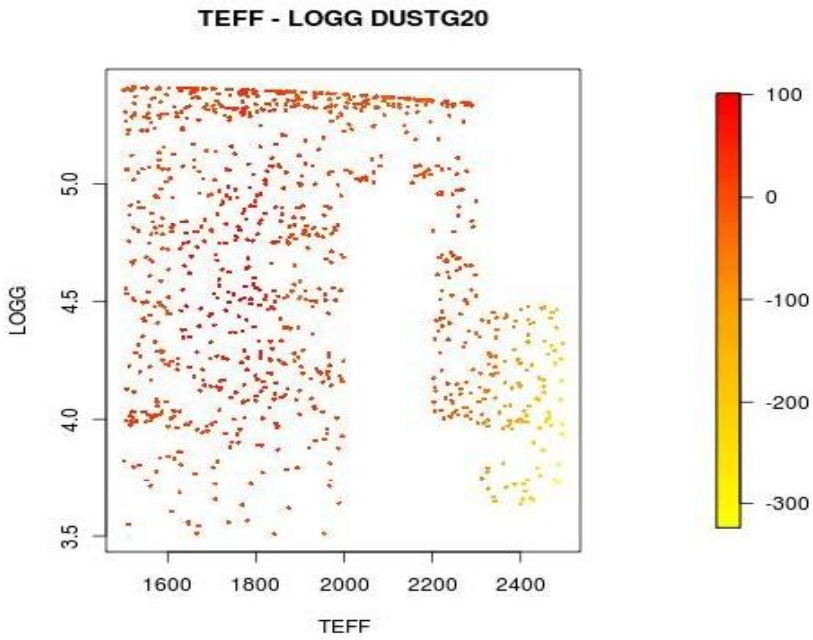


Figura 166 Gráfica de dispersión TEFF – LOGG para la predicción sobre DiffusionMaps GPs de TEFF para el conjunto de validación DUSTG20DM.

El clasificador es muy regular para todo el rango de temperaturas de los modelos DUST salvo para un caso puntual que habría que estudiar más al detalle

Sin embargo, existe un error de predicción focalizado en los 2400 ° K que afecta a la predicción de temperatura en el modelo DUST.

Habría que indagar el motivo por el cual aparece esta predicción tan mál clasificada

A priori parece que existe un problema con las predicciones en el rango final de las temperaturas del modelo DUST. Este error ha aparecido constantemente en otros clasificadores. Habría que estudiar si es motivos de reducción de dimensionalidad.

#### 3.2.2.3 Resultados para Transformador/Clasificador KPLS

Partiendo de el sistema de k-NN donde se fijaba el número de elementos a considerar. En los métodos de Kernel se fija una distancia h y "votan" todos los elementos que no disten más de ésta

Desde Weka la ejecución de KPLS presenta algunos inconvenientes. Se decide realizar la aplicación de la transformada y Clasificación mediante R.

Una parte buena de emplear R, es que como software estadístico permite evaluar y obtener el error cuadrático medio en función del número de coordenadas seleccionado.

Por lo tanto, para seleccionar el número adecuado de dimensiones a reducir mediante KPLS, se estudiará previamente este valor.

Para el conjunto de entrenamiento NOMarea, se obtienen los Errores cuadrados medios mostrados en la figura 167. En esta figura, para una transformada a 20 dimensiones, se obtienen los mejores resultados

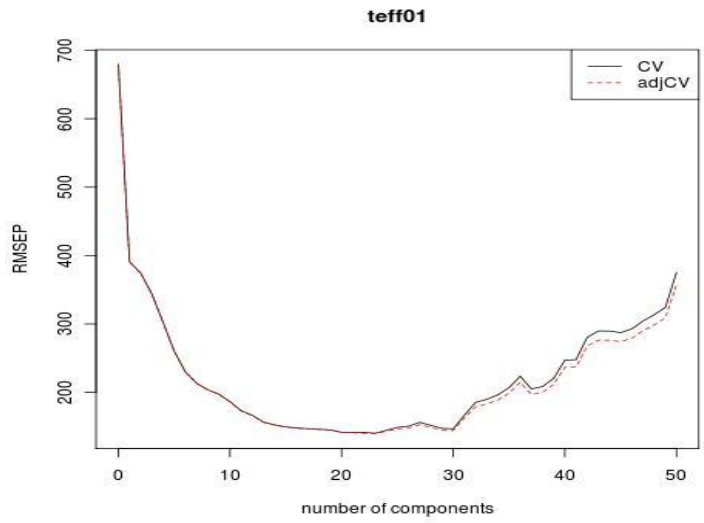


Figura 167 Evaluación del número de componentes de reducción aplicando transformada KPLS para el conjunto de entrenamiento NOVareal.

Sin embargo, si observamos la misma figura para el conjunto de validación RANG15 (véase figura 168), el menor error cuadrático medio se consigue para una transformada con al menos 50 componentes

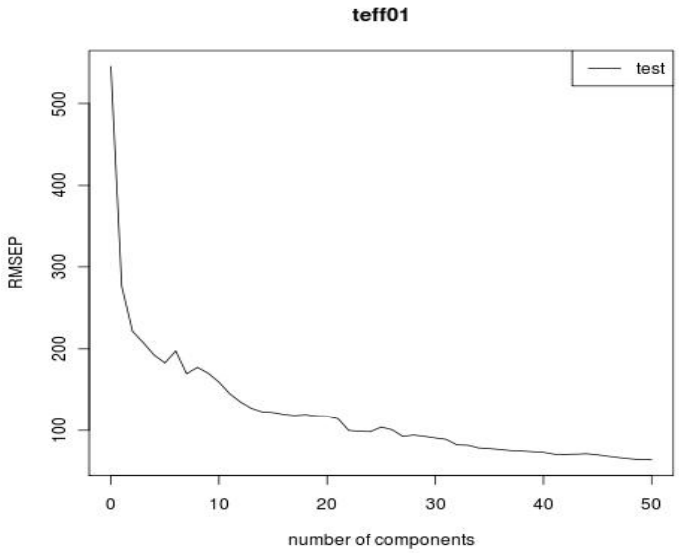


Figura 168 - Evaluación del número de componentes de reducción aplicando transformada KPLS para el conjunto de entrenamiento RANareal

Si observamos esta misma gráfica calculada para el conjunto de validación CONDG20area1 (véase figura 169) Para un número de dimensiones entre 10 y 20, se obtienen los mejores resultados

Sin embargo, si nos fijamos en el valor del error cuadrático, estamos hablando de errores medios de temperatura de 1000º K.

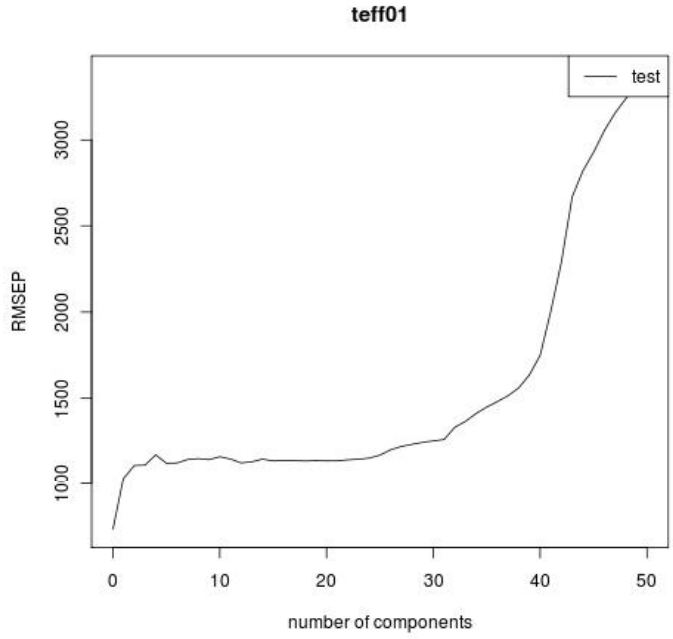


Figura 169 - Evaluación del número de componentes de reducción aplicando transformada KPLS para el conjunto de entrenamiento CONDG20area1

Se opta por emplear 20 componentes en la transformada KPLS.

Sin embargo, el sistema no da buenos resultados debido a que la variedad en el numero de dimensiones provoca que alguno de los conjuntos de entrenamiento no sean los adecuados

La tabla 46, 47 y 48 nos muestra el Error cuadrático medio en la predicción para los conjuntos de validación presentados en la tabla 7.

| | NOMarea1 | RANG15 |
|---|---|---|
| Root mean squared error | 212 | 169 |

Tabla 46. Error cuadrático medio para el conjunto de entrenamiento NOMarea1 y conjunto de validación RANG15

| | CONDG20 | CONDG20 moving | CONDG202Y | CONDG202Y moving |
|---|---|---|---|---|
| Root mean squared error | 1146 | 1097 | 1152 | 1104 |

Tabla 47. Error cuadrático medio para el conjunto de entrenamiento CONDG20area1, CONDG20area1moving, CONDG202Yarea1 e CONDG202Yarea1moving

| | DUSTG20 | DUSTG20 moving | DUSTG202Y | DUSTG202Y moving |
|---|---|---|---|---|
| Root mean squared error | 158 | 139 | 197 | 173 |

Tabla 48. Error cuadrático medio para los conjuntos de validación DUSTG20area1, DUSTG20area1moving, DUSTG202Yarea1 y DUSTG202Yarea1moving.

Las figuras 170 y 171 nos muestran respectivamente gráficas de dispersión de temperatura estimada para los valores de predicción sobre el conjunto de datos de validación CONDG20 empleando el clasificador KPLS.

Las figuras 172 y 173 nos muestran respectivamente gráficas de dispersión de temperatura estimada para los valores de predicción sobre el conjunto de datos de validación CONDG20movingav empleando el clasificador KPLS.

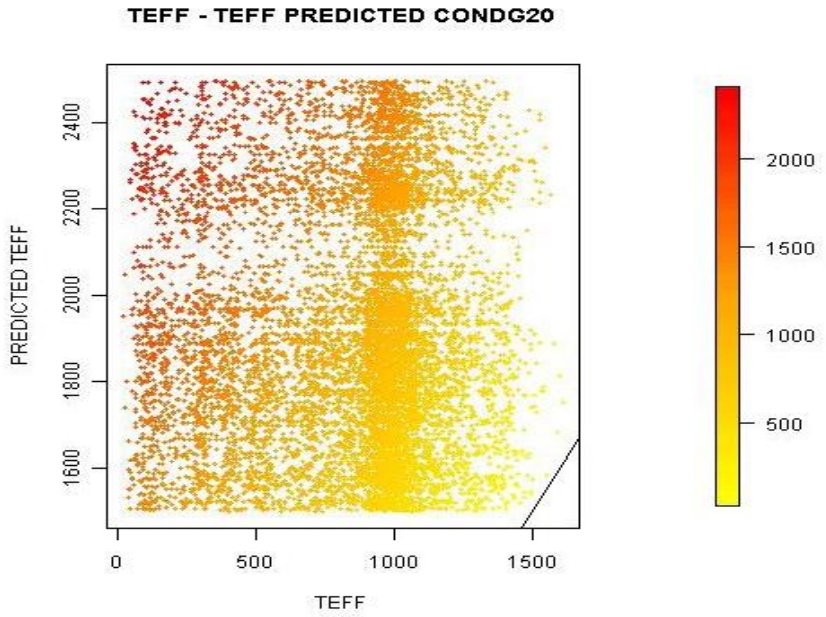


Figura 170 - gráfica de dispersión temperatura real frente a temperatura estimada por el clasificador KPLS para el conjunto de espectros de validación CONDG20

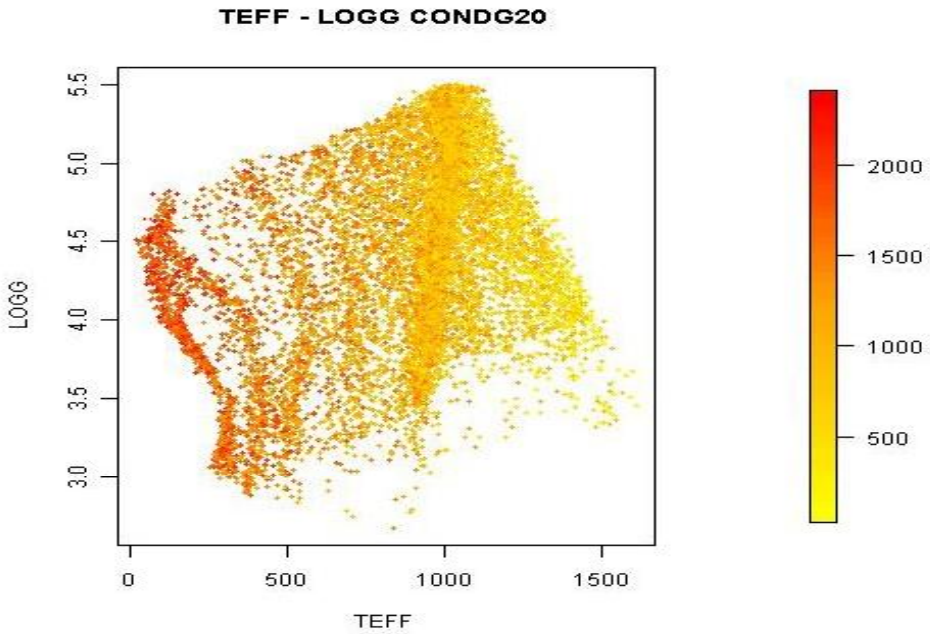


Figura 171 - gráfica de dispersión temperatura real frente a LOGG por el clasificador KPLS para el conjunto de espectros de validación CONDG20

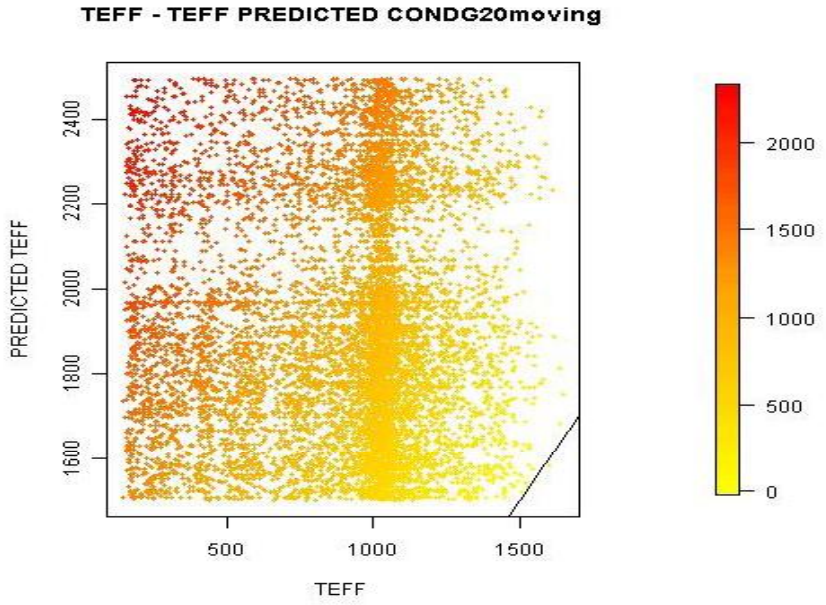


Figura 172 - gráfica de dispersión temperatura real frente a temperatura estimada por el clasificador KPLS para el conjunto de espectros de validación CONDG20moving

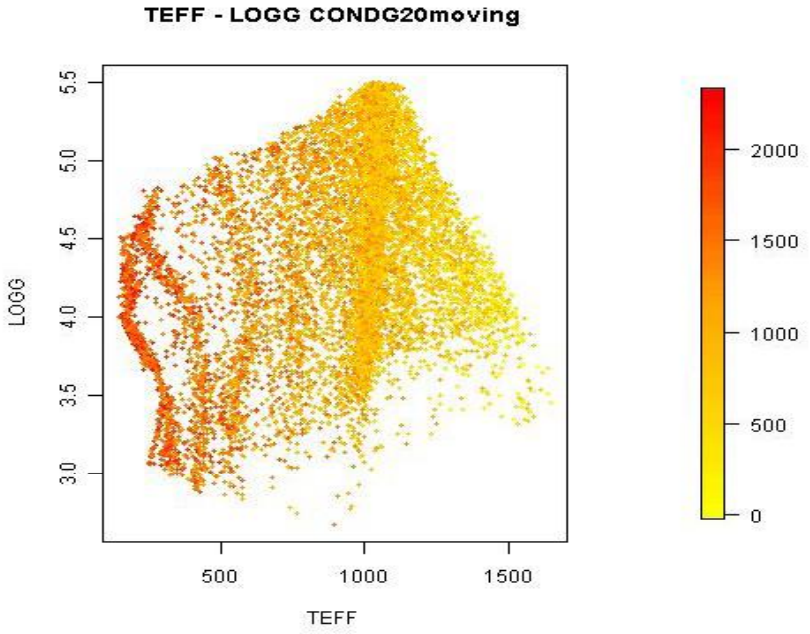


Figura 173 - gráfica de dispersión temperatura real frente a LOGG por el clasificador KPLS para el conjunto de espectros de validación CONDG20movingav

El clasificador no cumple correctamente con su función, hay que evaluar que problemas existen para el conjunto de datos de validación CONDG20 el sistema es muy malo

Ni si quiera la aplicación de suavizado sobre el conjunto de validación, se consiguen mejorar los resultados.

Para el rango de temperaturas desde 100° a 1600 ° K el sistema predice temperaturas efectivas muy por encima de su valor real, en realidad ya se había observado este comportamiento al visualizar la curva de errores cuadráticos en función del numero de componentes (figura 169).

Sin embargo, para el rango de temperaturas de DUSTG20 (por encima de 1600°K) el sistema al menos realiza una predicción más aproximada.

Se puede observar en las siguientes figuras:

Las figuras 174 y 175 nos muestran respectivamente gráficas de dispersión de temperatura estimada para los valores de predicción sobre el conjunto de datos de validación DUSTG20 empleando el clasificador KPLS

Las figuras 176 y 177 nos muestran respectivamente gráficas de dispersión de temperatura estimada para los valores de predicción sobre el conjunto de datos de validación DUSTG20moving empleando el clasificador KPLS.

Téngase en cuenta que el degradado de color para las cuatro figuras, nos muestra el error en la predicción de la temperatura.

Los valores amarillos nos muestran los valores mínimos de desviación, generalmente temperaturas que no han alcanzado el valor real, mientras que los valores rojos nos muestran los valores máximos de desviación, generalmente marcados por temperaturas predichas por encima de su valor real.

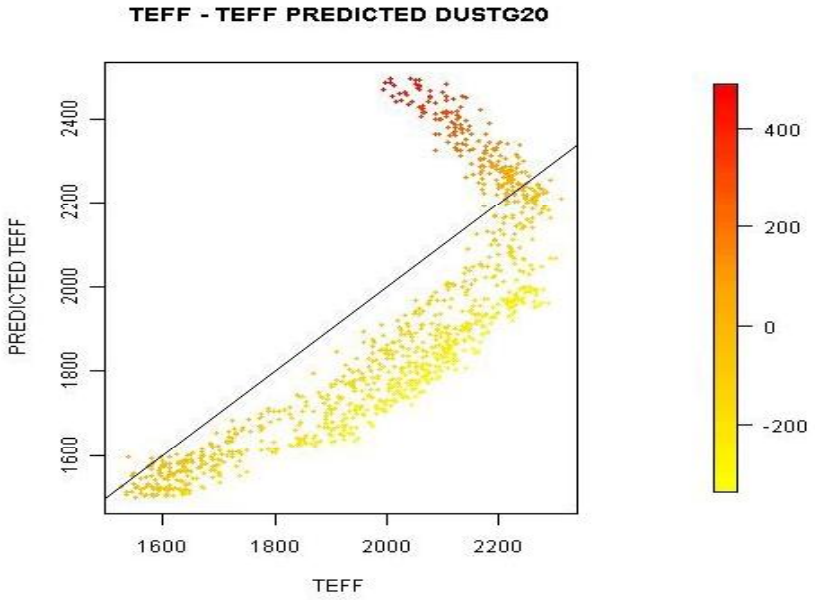


Figura 174 - gráfica de dispersión temperatura real frente a temperatura estimada por el clasificador KPLS para el conjunto de espectros de validación DUSTG20

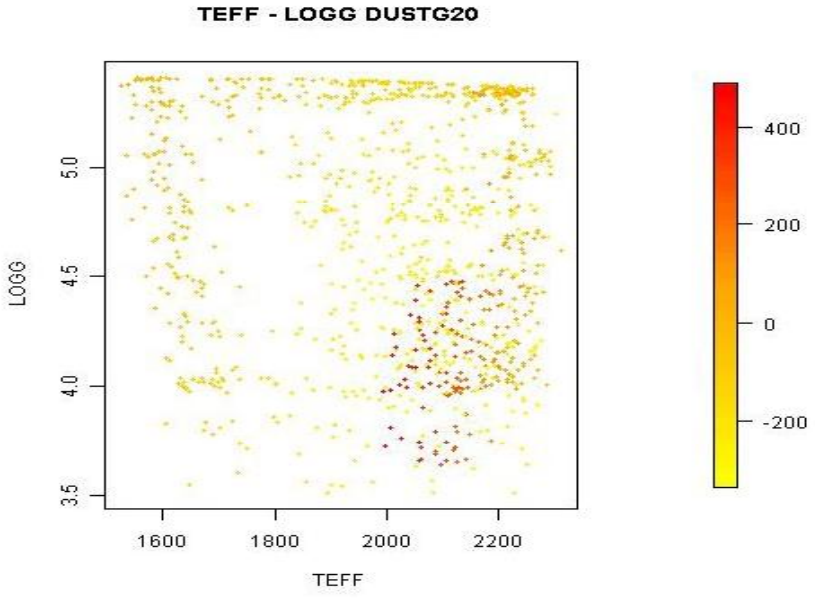


Figura 175 - gráfica de dispersión temperatura real frente a LOGG por el clasificador KPLS para el conjunto de espectros de validación DUSTG20

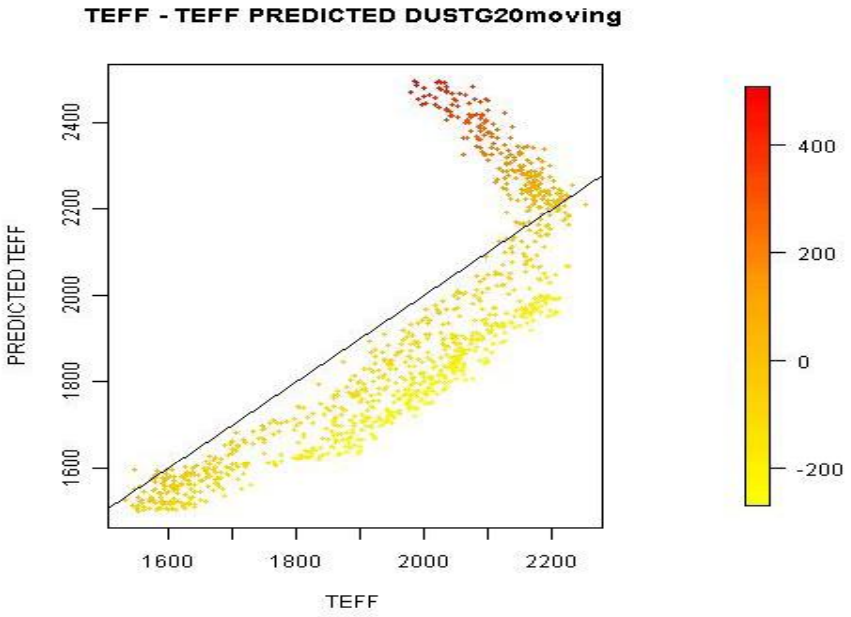


Figura 176 - gráfica de dispersión temperatura real frente a temperatura estimada por el clasificador KPLS para el conjunto de espectros de validación DUSTG20moving

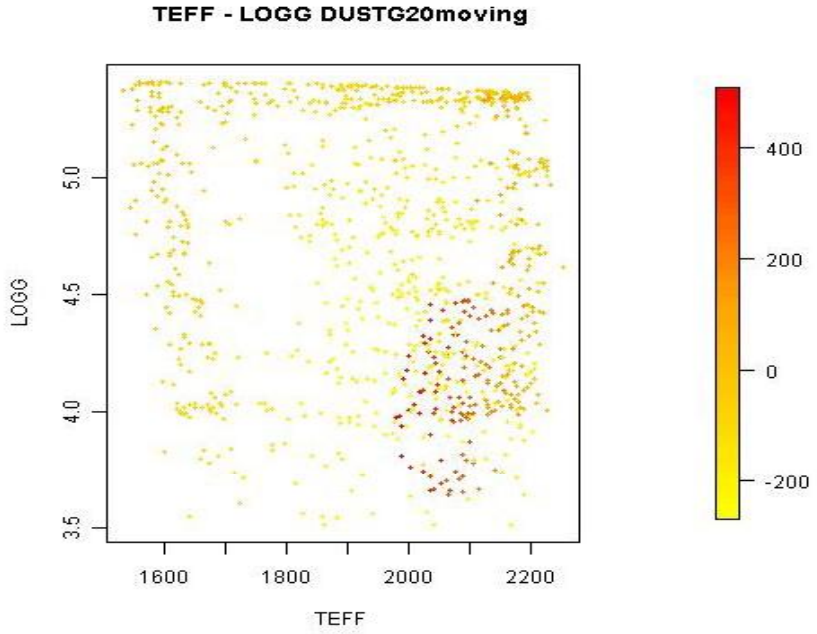


Figura 177 - gráfica de dispersión temperatura real frente a LOGG por el clasificador KPLS para el conjunto de espectros de validación DUSTG20movingav

El clasificador se comporta mejor que para los modelos COND

Para temperaturas por debajo de 2000ºK, el sistema propone temperaturas efectivas por debajo de las reales

A partir de 2000ºK, el clasificador realiza predicciones muy ambiguas tanto por encima como por debajo de la temperatura efectiva real

Los errores por encima de la temperatura real se predicen para espectros cuyo valor del logaritmo de la gravedad se encuentra por debajo de 4.5

#### 3.2.2.4 Conclusiones Parciales

Viendo el resultado del error cuadrático, no hubiese echo falta la representación mediante gráficas de dispersión para tener claro que KPLS mediante 20 componentes no ofrece un buen resultado

Para temperaturas por debajo de 1600ºK, el sistema se comporta realmente mal, obteniendo predicciones muy malas, inservibles para nuestro estudio.

El sistema se comporta más coherente en predicciones por encima de los 1600 ºK. Al menos en este rango de temperaturas sigue un patrón común, predicciones por debajo de la real

Se debe analizar durante la ejecución de la tesis doctoral, el motivo del pésimo comportamiento de KPLS con los conjuntos de datos de validación con ruido empleados.

### 3.2.3 Experimentación con conjuntos de entrenamiento con ruido

Tras conversaciones con los otros grupos de DPAC, se comenta que se obtienen mejores resultado si al conjunto de entrenamiento se le añade ruido

Se ha realizado una pequeña prueba de predicción sobre los 3 primeros clasificadores sin reducción de dimensionalidad para comprobar el resultado

Al menos el error cuadrático medio es inferior. Deberia realizarse un análisis más al detalle profundizando en diferentes modelos COND y DUST

También seria aconsejable realizar Inferencias Bayesianas o T-Student para determinar si realmente son mejores clasificadores.

En la tabla 48 se muestra el resultado de este estudio previo realizado

| MODELO | | NOMarea1 | RANG15 |
|---|---|---|---|
| Area 1 + KNN | Correlation coefficient | 0.9929 | 0.9953 |
| | Mean absolute error | 37.234 | 34.2502 |
| | Root mean squared error | 80.5666 | 47.1759 |
| | Relative absolute error | 6.4547 % | 5.4568 % |
| | Root relative squared error | 11.3631 % | 6.6224 % |
| Area1 + SMO | Correlation coefficient | 0.9874 | 0.9988 |
| | Mean absolute error | 68.0928 | 18.6237 |
| | Root mean squared error | 110.1875 | 25.9345 |
| | Relative absolute error | 11.804[illegible] % | 4.1381 % |
| | Root relative squared error | 16.2266 % | 4.8417 % |
| Area 1 [illegible] G25 | Correlation coefficient | 0.9662 | 0.9981 |
| | Mean absolute error | 128.1305 | 22.5468 |
| | Root mean squared error | 175.4508 | 33.5943 |
| | Relative absolute error | 22.2119 % | 5.4262 % |
| | Root relative squared error | 25.8376 % | 6.5306 % |

Tabla 48. Resultado de la clasificación entrenando con NOM con RUIDO

Cabe decir que durante la ejecución de la tesis Doctoral se profundizará también en este aspecto llegando a concluir e intentar razonar el motivo de esa mejoria que hoy por hoy no encontramos justificación.

## 4. Conclusiones

A la vista de los resultados y comportamientos de los diferentes clasificadores, si bien no se puede evaluar y decidir sobre cuál clasificador se comporta mejor hasta que no se realice un estudio más detallado, por ejemplo infiriendo los parámetros mediante técnicas bayesianas o aplicando el T-Student sobre determinados clasificadores

Existe una serie de comportamientos que se debe profundizar.

Por ejemplo, que los conjuntos de datos de validación para menos de 500 ° Kelvin deben revisarse, por algún motivo, existe un problema en la interpolación de datos para ese rango específico, que hace que el sistema se comporte de una forma muy singular, escalonando las predicciones

En el análisis inicial de los datos, en las figuras 14, 15 16 y 17 se observa como los atributos de los espectros están muy correlacionadas, de forma que a priori, la reducción de dimensionalidad debería mejorar los resultados de cualquier clasificador sin reducción.

Esta mejoría no es muy aparente en el estudio realizado, es muy posible que la determinación de aplicar PCA con el criterio de cobertura del 95% de la varianza no es muy acertado ya que se reduce la dimensionalidad del espectro de 180 a 6 atributos.

Por los estudios que se están realizando en otras Unidades del DPAC, se observa que quizás la mejor reducción de dimensionalidad se encuentre a 20 y 30 atributos

Hay que profundizar más sobre la mejora de PCA en los clasificadores mostrados. Por supuesto, este estudio formará parte de la ya comentada tesis doctoral.

Un análisis rudo de los resultados, sin gran significado hasta que se valide correspondientemente como se ha comentado anteriormente, nos muestran como, para magnitud aparente G20 y sin

reducción de dimensionalidad, el clasificador que parece comportase mejor para temperaturas entre 100 y 1600 Kelvin (modelos COND) son las máquinas de vectores soporte

Para las temperaturas entre 1500 y 2500 Kelvin (modelos DUST) sobre los modelos de magnitud aparente 20, KNN para 5 vecinos cercanos, aplicando previamente la reducción y transformación de los conjuntos de datos mediante Análisis de Componentes Principales, es clasificador que mejores resultados muestra.

La experimentación posterior demuestra que no se consiguen mejorar estos resultados aplicando la transformación mediante Partial Least Square

Es interesante profundizar más en los Mapas de Difusión ya que aportan sistemas de predicción muy homogeneos y continuos para las diferentes temperaturas efectivas.

En etapas intermedias de la misión en las que el número de observaciones de una estrella dada sea inferior al total (70 observaciones en promedio) la relación señal/ruido será muy inferior por lo que el sistema diseñado deberá ser especialmente robusto.

El sistema predictivo k-NN con suavizado moving average previo sobre los conjuntos de datos facilitados, aparentemente obtiene mejor resultado indiferentemente de la teff esperada (tanto para los modelos COND como DUST)

## 5. Investigación Futura

Dados los resultados obtenidos, surgen nuevos retos para poder intentar alimentar los modelos predictivos evaluados sobre datos reales

Como se ha hecho referencia constantemente, se deben de abordar determinadas técnicas de inferencia técnicas bayesianas o aplicando el T-Student sobre determinados clasificadores para poder realizar una adecuada toma de decisiones

Este tipo de estudio más detallado no es objeto del Trabajo fin de máster sino de su investigación futura que se desarrollará mediante la ejecución de una tesis doctoral.

Tras conversaciones con los otros grupos de DPAC, y como he ha observado en el apartado 3.2.3, se obtienen mejores resultado si al conjunto de entrenamiento se le añade ruido

Por lo tanto, existen evidencias que nos obligan a profundizar más sobre los clasificadores y los conjuntos de entrenamiento. Alguna parte de la investigación no podrá llevarse a cabo hasta que la misión GAIA se ponga operativa.

Por las conclusiones comentadas en el apartado anterior, referentes a la reducción de dimensionalidad mediante Componentes Principales, Mapas de Difusión o Partial Least Square, y por la observación de los espectros en las figuras 14, 15 16 y 17, debería existir una mejoría que no es aparente en el estudio realizado.

Es muy posible, por ejemplo, que la determinación de aplicar PCA con el criterio de cobertura del 95% de la varianza no haya sido muy acertado ya que se reduce la dimensionalidad del espectro de 180 a 6 atributos.

Por los estudios que se están realizando en otras Unidades del DPAC, se observa que quizás la mejor reducción de dimensionalidad se encuentre a 20 y 30 atributos

Hay que profundizar más sobre la mejora de PCA en los clasificadores mostrados.

Para ello, y una vez desarrollada esta fase de experimentación descrita en el presente trabajo, se pretende dar continuidad a la investigación mediante el desarrollo de una tesis doctoral la cual se centrará:

Inicialmente en el estudio de los clasificadores empleando conjuntos de entrenamiento con ruido. Para ello se deberá estudiar la magnitud del ruido a generar en ese conjunto de entrenamiento. Profundizar sobre el entrenamiento con ruido.

- Estudio de la reducción de dimensionalidad.

Inferencia de parámetros mediante técnicas bayesianas o aplicando el T-Student sobre determinados clasificadores para poder realizar una adecuada toma de decisiones

Implementación en Java de la solución/soluciones finales dentro del entorno del consorcio DPAC

- Validación de los sistemas propuestos con conjuntos de entrenamiento reales facilitados por GAIA en las etapas intermedias de la misión, así como al final de la misma

- Mantenimiento y mejora de la aplicación y algoritmos desarrollados durante el tiempo que dura la misión GAIA.

## 6. Bibliografía

*Especifica de los conjunto de datos usados y sobre misión GAIA:*

[1] ALLARD, FRANCE. The limiting effects of DUST in Brown Dwarf Model Atmospheres. Centre de Recherche Astronomique de Lyon (CRAL), Ecole Normale Superieure de Lyon October 2000

[2] ALLARD, FRANCE, G. CHABRIER, I. BARAFFE AND P. HAUSCHILDT - Evolutionary models for very low-mass stars and brown dwarfs with dusty atmospheres - The Astrophysical Journal, 542 : 464è472, 2003

[3] ALLARD, FRANCE, G. CHABRIER, I. BARAFFE AND P. HAUSCHILDT - Evolutionary models for cool dwarfs and extrasolar giant planets the case of hd 209458 - The Astrophysical Journal – 2001

[4] IAN H. WITTEN AND EIBE FRANK - Data Mining: Practical Machine Learning Tools and Techniques, Second Edition - Department of Computer Science University of Waikato

[5] MUÑOZ LUJAN ,CARLOS – Proyecto Fin de Carrera - Implementación de métodos clasicos de Regresión – 2010

[6] R DEVELOPMENT CORE TEAM - R: A Language and Environment for Statistical Computing - Version 2.10.1 (2009-12-14)

[7] VARIOS -CU8- Probability Density Estimators Optimization and Multivariable Stadistical Test Implementation for "GAIA" Mission. - The American Astronomical Society, July 20 10

[8] VARIOS. CU8: Astrophysical Parameters Algorithmic Structure

*General sobre métodos y modelos matemáticos implementados en el proyecto:*

[9] ABRAHAM , C., BIAU . G., AND CADRE . B. On the kernel rule for function classification. Annals of the Institute of Statistical Mathematics 58, 3 (2006), 619–633

[10] ACI . M , INAN . C., AND AVCI , M. A hybrid classification method of k nearest neighbour, Bayesian methods and genetic algorithm. Expert Systems with Applications (2009)

[11] BAÍLLO . A , CUESTA -ALBERTOS , J., AND CUEVAS , A. Supervised classification for a family of Gaussian functional models. Scandinavian Journal of Statistics (under revision) (2010)

[12] BARKER . M , AND RAYENS , W. Partial least squares for discrimination. Journal of Chemo metrics 17, 3 (2003), 166 – 173

[13] BENKO , M., HÄRDLE . W., AND KNEIP, A. Common functional principal components Annals of Statist 37 (2009), 1–34

[14] BERLINET, A., BIAU . G., AND ROUVIERE , L. Functional Supervised Classification with Wavelets. In Annales de l'ISUP (2008), vol. 52, Institut de statistique de l'Université de Paris, pp. 61–80.

[15] BIAU . G., BUNEA , F., AND WEGKAMP, M. Functional classification in Hilbert spaces IEEE Transactions on Information Theory 51, 6 (2005), 2163–2172

[16] BOULESTEIX , A.-L., AND STRIMMER . K. Partial least squares: a versatile tool for the analysis of high-dimensional genomic data. Briefings in Bioinformatics 8, 1 (2007), 32–44.

[17] C.E. RASMUSSEN Y C K I. WILLIAMS. Gaussian Processes for Machine Learning. The MIT Press. 2006. ISBN 026218253X. c 2001

[18] CUEVAS , A., FEBRERO , M., AND FRAIMAN , R. Robust estimation and classification for functional data via projection-based depth notions. Computational Statistics 22, 3 (2007), 481–496.

[19] FORT, G., AND LAMBERT-LACROIX , S. Classification using partial least squares with penalized logistic regression. Bioinformatics 21 (2005), 1104–1111.

[20] PREDA , C. Regression models for functional data by reproducing kernel Hilbert spaces methods. Journal of statistical planning and inference 137, 3 (2007), 829–840.

[21] TORRECILLA NOGUERALES, JOSÉ LUIS. Análisis de Datos Funcionales, Clasificación y Selección de Variables - 2010

[22] VAPNIK , V. N. Statistical Learning Theory. Wiley-Interscience, 1998

[23] VEGA VILCA , J. C. Generalizaciones de mínimos cuadrados parciales con aplicación en clasificación supervisada. PhD thesis, Universidad de Puerto Rico, 2004