

UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

Máster Universitario En Inteligencia Artificial
Avanzada: Fundamentos, Métodos y Aplicaciones

Especialidad de Sistemas Inteligentes de
Diagnóstico, Planificación y Control

TRABAJO FIN DE MÁSTER

Estimación de la temperatura y gravedad de estrellas del
archivo de COROT a partir de espectros FLAMES

Gonzalo León Manzano
Septiembre de 2011

Índice de contenido

1	Introducción y objetivos.....	6
2	Estado del arte.....	7
	2.1 Determinación de diversos parámetros atmosféricos estelares a partir de espectros de ELODIE.....	7
	2.1.1 Preparación del espectro.....	7
	2.1.2 Comparación del espectro con el conjunto de referencia.....	8
	2.1.3 Resultados.....	9
	2.2 El algoritmo MATISSE.....	9
	2.3 El método “piecewise PLS”.....	10
3	Solución propuesta.....	11
	3.1 Entorno y herramientas de desarrollo.....	12
	3.2 Preparación de los datos.....	13
	3.2.1 Interpolación.....	13
	3.2.2 Eliminación del ruido.....	16
	3.2.3 Sustracción del continuo.....	19
	3.2.4 Eliminación de rayos cósmicos.....	22
	3.2.5 Estimación de la relación señal-ruido.....	23
	3.2.6 Corrección del desplazamiento Doppler.....	27
	3.2.7 Normalización.....	31
	3.3 Minería de datos.....	32
	3.3.1 Análisis de componentes principales y máquinas de vectores soporte.....	32
	3.3.1.1 Análisis de componentes principales (PCA).....	32
	3.3.1.2 Primeros experimentos de regresión con máquinas de vectores soporte (ϵ -SVM).....	33
	3.3.1.3 Clasificador dicotómico.....	38
	3.3.1.4 Experimentos finales de regresión con máquinas de vectores soporte (ϵ -SVM).....	43
	3.3.2 Aplicación de los modelos predictivos a FLAMES	49
4	Otros experimentos.....	50
	4.1 Fusión de espectros en FLAMES.....	50
	4.2 K-vecinos más próximos.....	52
	4.3 Diffusion maps y SVM.....	56
	4.4 Regresión Partial Least Squares.....	63
	4.5 Entrenamiento con ELODIE.....	67
	4.6 Entrenamiento con TLUSTY.....	68
5	Comparación de resultados.....	70
6	Conclusiones y mejoras.....	72
	Apéndice I. Fórmulas de medida del error.....	73
	Bibliografía.....	74

Índice de ilustraciones

Ilustración 1: Ejemplo de los puntos originales e interpolados de flujo en un modelo de Kurucz....	16
Ilustración 2: Proceso de eliminación del ruido a través de wavelets.....	17
Ilustración 3: Efecto de la supresión del ruido en la estrella 100552898 de FLAMES I.....	18
Ilustración 4: Efecto de la supresión del ruido en la estrella 00001 de ELODIE.....	18
Ilustración 5: Sustracción del continuo en un modelo de Kurucz en el rango azul.....	20
Ilustración 6: Sustracción del continuo de un modelo de Kurucz en el rango rojo.....	20
Ilustración 7: Sustracción del continuo de la estrella 100991167 de FLAMES en el rango azul.....	21
Ilustración 8: Sustracción del continuo en la estrella 100919104 de FLAMES en el rango rojo.....	21
Ilustración 9: Eliminación de rayos cósmicos en la estrella 101091412 (rango azul de FLAMES I).	22
Ilustración 10: Eliminación de rayos cósmicos en la estrella 110827797 (2010-12-13 08h02) Rango rojo, FLAMES II.....	23
Ilustración 11: Espectro del FLAMES en el rango rojo con $SNR < 0.5$	24
Ilustración 12: Espectro del FLAMES en el rango azul con $0.5 < SNR < 1$	24
Ilustración 13: Espectro FLAMES en el rango rojo con $1 < SNR < 2$	25
Ilustración 14: Espectro FLAMES en el rango azul con $3 < SNR < 4$	25
Ilustración 15: Espectro FLAMES en el rango rojo con $4 < SNR < 6$	26
Ilustración 16: Espectro FLAMES en el rango azul con $6 < SNR < 8$	26
Ilustración 17: Espectro FLAMES en el rango azul con $SNR > 8$	27
Ilustración 18: Distribución de estrellas de FLAMES I en PC1 y PC2.	28
Ilustración 19: PC1 vs. PC2 de FLAMES I coloreados según su desplazamiento Doppler.....	29
Ilustración 20: Estrellas de FLAMES I después de la corrección del desplazamiento doppler.....	30
Ilustración 21: Varios espectros de FLAMES II antes de la normalización.....	31
Ilustración 22: Espectros de FLAMES II después de la normalización.....	31
Ilustración 23: Varianza acumulada por número de componentes principales.....	33
Ilustración 24: Temperatura efectiva vs. real utilizando PCA+SVM con el kernel RBF.....	37
Ilustración 25: Componentes principales 2 y 8 en los modelos de Kurucz.	42
Ilustración 26: Componentes principales 2 y 8 en las estrellas de ELODIE.	43
Ilustración 27: Temperatura efectiva predicha vs. real para estrellas por debajo de 10.000 K, rango azul, utilizando PCA + SVM.....	47
Ilustración 28: Gravedad predicha vs. real para estrellas por debajo de 10.000 K, rango azul, utilizando PCA + SVM.....	47
Ilustración 29: Temperatura efectiva predicha vs. real para estrellas por debajo de 10.000 K, rango rojo, utilizando PCA + SVM.....	48
Ilustración 30: Gravedad predicha vs. real para estrellas por debajo de 10.000 K, rango rojo, utilizando PCA + SVM.....	48
Ilustración 31: Teff y $\log(g)$ predichos para FLAMES I.	49
Ilustración 32: Teff y $\log(g)$ predichos para FLAMES II.....	50
Ilustración 33: Temperatura efectiva predicha frente a gravedad predicha en FLAMES I, para el rango azul (en color azul), en rango rojo (en color rojo) y las estrellas fundidas en el rango rojo (en color verde). Se utilizó PCA + SVM para la predicción.....	51
Ilustración 34: Espectro fundido anómalo en FLAMES I en color negro, y los espectros originales de donde proviene en otros colores, todos de la misma estrella: 100740166.....	51
Ilustración 35: Temperatura efectiva predicha vs. real para estrellas por debajo de 10.000 K, rango azul, utilizando 7 vecinos más próximos.....	54
Ilustración 36: Gravedad predicha vs. real para estrellas por debajo de 10.000 K, rango azul,	

utilizando 7 vecinos más próximos.....	55
Ilustración 37: Temperatura efectiva predicha vs. real para estrellas por debajo de 10.000 K, rango rojo, utilizando 7 vecinos más próximos.....	55
Ilustración 38: Gravedad predicha vs. real para estrellas por debajo de 10.000 K, rango rojo, utilizando 7 vecinos más próximos.....	56
Ilustración 39: Coordenadas de difusión 1 y 2.	59
Ilustración 40: Coordenadas de difusión 1 y 3.	59
Ilustración 41: Coordenadas de difusión 1 y 2 para estrellas menores de 10000K.	60
Ilustración 42: Temperatura real vs. predicha utilizando Diffusion Maps y SVM, en el rango azul para estrellas < 10013 K.....	62
Ilustración 43: log(g) real vs. predicha utilizando Diffusion Maps y SVM, en el rango azul, para estrellas menores de 10013 K.....	63
Ilustración 44: RMSE en función del número de componentes en PLS para Kurucz en el rango azul	64
Ilustración 45: RMSE en función del número de componentes en PLS para Kurucz en el rango rojo	65
Ilustración 46: RMSE vs. número de componentes en PLS para ELODIE, rango azul.....	66
Ilustración 47: log(g) predicho vs. real en el rango azul, utilizando PLSR	66
Ilustración 48: Predicción de Teff y log(g) en FLAMES I, entrenando con ELODIE.....	67
Ilustración 49: Predicción de Teff y log(g) en FLAMES II, entrenando con ELODIE.....	68
Ilustración 50: Distribución de las temperaturas de los modelos de Kurucz y TLUSTY.....	69

Índice de tablas

Tabla 1: Diferencias entre los conjuntos de datos de Kurucz y de FLAMES I.....	14
Tabla 2: Errores encontrados por kernel y rango.....	36
Tabla 3: Errores del conjunto completo y el conjunto con temperaturas ≤ 20.000	37
Tabla 4: FPR y FNR en el rango azul para diferentes algoritmo de clasificación binaria.	38
Tabla 5: Clasificación de estrellas frías (<10000 K) y calientes (≥ 10000 K).	41
Tabla 6: Clasificación de estrellas frías (≤ 10013 K) y calientes (>10013 K).	42
Tabla 7: Resultados con 30 PC, validación cruzada, conjuntos completos.	44
Tabla 8: Resultados con 30 PC, validación cruzada, estrellas y modelos < 10000 K.	44
Tabla 9: Resultados con PC seleccionados , validación cruzada, estrellas y modelos < 10000 K.....	45
Tabla 10: Resultados con PC seleccionados , validación ELODIE, estrellas y modelos <10000 K. $\sigma=0,0625$	46
Tabla 11: Resultados sobre conjunto de test ELODIE, utilizando 7 vecinos más próximos en Kurucz, para los conjuntos completos de estrellas.....	53
Tabla 12: Resultados sobre conjunto de test ELODIE, utilizando 7 vecinos más próximos en Kurucz, para estrellas y modelos < 10013 K.	53
Tabla 13: Resultados sobre conjunto de test ELODIE, usando Diffusion Maps y SVM.....	60
Tabla 14: Resultados sobre conjunto de test ELODIE, utilizando Diffusion Maps y SVM.....	61
Tabla 15: Resultados sobre conjunto de test ELODIE, utilizando Diffusion Maps y SVM.	62
Tabla 16: Resultados de aplicar regresión PLS en el rango azul sobre el conjunto de ELODIE. La validación se hizo sobre ELODIE.....	65
Tabla 17: Resultados sobre TLUSTY, utilizando las 30 primeras componentes principales y SVM, con validación cruzada de 10 subconjuntos sobre TLUSTY.....	69
Tabla 18: Resultados sobre ELODIE, utilizando las 30 primeras componentes principales y SVM, con validación cruzada sobre el conjunto TLUSTY.....	70
Tabla 19: Comparación entre métodos predictivos con el mismo conjunto de validación que de entrenamiento.....	71
Tabla 20: Comparación entre métodos predictivos cuyo conjunto de validación difiere del de entrenamiento.....	71

1 Introducción y objetivos

El presente documento constituye la memoria del trabajo final de iniciación a la investigación para el *Máster de Inteligencia Artificial Avanzada*, en su especialidad de *Sistemas Inteligentes de Diagnóstico, Planificación y Control*.

El área de conocimiento en el que la investigación se halla enmarcada es la Minería de Datos, cuyo objetivo es el descubrimiento de patrones en grandes volúmenes de datos. El conjunto de datos del que se quiere extraer conocimiento procede de la Astronomía, en concreto, son los espectros de un conjunto de estrellas de los que se pretende estimar diversos parámetros físicos.

Es necesario exponer previamente algunos aspectos fundamentales sobre Astronomía antes de exponer los objetivos del trabajo. Se llama espectro de una fuente de radiación al conjunto ordenado de radiaciones emitido por la fuente en las distintas longitudes de onda de la radiación. Cualquier estrella es una fuente de radiación, y los espectros de las estrellas (incluido el Sol) presentan una particularidad: en ellos se aprecian descensos y aumentos bruscos en determinadas longitudes de onda. Los descensos son debidos a que la luz es absorbida por los elementos químicos de la atmósfera de la estrella solamente en determinadas longitudes de onda, y redirigidos en todas direcciones. Por tanto, la radiación original es privada de parte de sus fotones en unas longitudes de onda que están determinadas por los elementos químicos presentes en la atmósfera de la estrella, lo que provoca que en su espectro aparezcan rayas oscuras, llamadas *líneas de absorción* o *líneas de Fraunhofer*.

Estas y otras peculiaridades de los espectros de las estrellas han sido aprovechadas por los científicos para clasificar las estrellas; por ejemplo, se han deducido diversos parámetros físicos de las estrellas como la temperatura superficial, la gravedad, y otra información acerca de su composición química. Sin embargo, la espectrografía astronómica dista mucho de ser una ciencia exacta. No existe una forma automática y sencilla para clasificar de forma exacta la estrella, y el estudio de una estrella necesita del conocimiento de un experto que analice detalladamente su espectro, la variabilidad en su flujo lumínico, y otras características, conocimiento que ha de ser adquirido tras años de estudio.

Es en este punto donde la Minería de Datos puede contribuir a la estimación de los ya mencionados parámetros físicos de las estrellas. Nótese que el interés por esta disciplina surge de la capacidad de los ordenadores (relativamente reciente) de realizar cálculos muy rápidamente; sin un ordenador, nadie se plantearía tratar de extraer información de un conjunto muy grande de datos. Y los espectros de las estrellas pueden ser conjuntos grandísimos de datos, no sólo por la cantidad de estrellas que son susceptibles de ser analizadas, sino por la cantidad de longitudes de onda que puede contener un espectro.

En realidad, el término Minería de Datos forma parte de un proceso más amplio, conocido como descubrimiento de conocimiento en bases de datos (*Knowledge Discovery from Databases, KDD*), si bien, muchas veces se ha acuñado el primero de estos términos para referirse al proceso completo. Las fases de este proceso son:

1. Integración y recopilación de datos, donde se determinan las fuentes de información, y se integran en un solo almacén de datos en un formato común.
2. Selección, limpieza y transformación, en la que se eliminan o corrigen los datos incorrectos o incompletos, seleccionando los más relevantes.
3. Minería de datos, donde se trata de producir un modelo cuyos datos de entrada sean los

datos seleccionados en el punto anterior, y la salida es el conocimiento que se pueden extraer de dichos datos.

4. Evaluación e interpretación donde se evalúa el modelo generado, y se analiza en el contexto donde va a ser utilizado.
5. Difusión y uso de modelos. En este punto, el modelo se distribuye para su utilización, y se vigila en una evaluación continua para evitar que se degrade con la llegada de nuevos datos que no haya contemplado el modelo original.

A los dos primeros puntos se les suele englobar en una sola fase llamada *preparación de datos*.

En este contexto, el objetivo del presente trabajo es el de deducir, utilizando técnicas de Minería de Datos, la temperatura efectiva y la gravedad de la estrellas (el conocimiento a extraer) a partir de los espectros de las estrellas (los datos de entrada). Además, dado que los parámetros físicos son datos numéricos, la tarea predictiva es la regresión.

En el capítulo 2 de este documento se analizarán los últimos progresos realizados en este aspecto. El capítulo 3 describe la solución propuesta. El capítulo 4 presenta otros experimentos realizados cuyos resultados fueron menos afortunados. El capítulo 5 es un estudio comparativo de los resultados obtenidos con respecto a otros trabajos. El capítulo 6 presenta las conclusiones generales del trabajo realizado y propone futuras mejoras. Los últimos capítulos están dedicados a definir las medidas de errores utilizadas y la bibliografía utilizada.

2 Estado del arte

2.1 Determinación de diversos parámetros atmosféricos estelares a partir de espectros de ELODIE

ELODIE fue un espectrógrafo echelle utilizado en el telescopio del Observatorio de la Alta Provenza entre los años 1993 y 2006. Uno de los estudios (1) que se llevaron a cabo fue un método para la estimación la temperatura efectiva, el logaritmo de la gravedad y la metalicidad $[Fe/H]$. Está fundamentado en la comparación del espectro de la estrella que se desea estudiar con una librería de espectros de estrellas de referencia. La librería de estrellas está compuesta por 211 estrellas estudiadas previamente en el rango de temperaturas de los 4000K a los 6300K, y cuyos parámetros atmosféricos estimados son de una alta fiabilidad. El sistema fue implementado en una aplicación llamada TGMET. Este catálogo de estrellas fue ampliado en versiones posteriores. La colección referida en este apartado fue una de las primeras, y contiene un conjunto mucho más reducido de estrellas de las tiene la última versión (la 3.1), que es la que se utiliza en otros apartados de este documento.

2.1.1 Preparación del espectro

Los espectros de las estrellas de referencia fueron sometidos a varias transformaciones para eliminar las características que no son propias de la estrella. Estas son las transformaciones:

Calibración de longitud de onda

Consiste en una interpolación mediante un polinomio de Chebyshev para calcular el valor en cada longitud de onda a partir de los valores de píxeles obtenidos directamente del espectrógrafo.

Corrección de píxel y enderezamiento por cada orden

Cada píxel del espectrógrafo tiene una sensibilidad diferente, por lo que se corrigió cada píxel para homogeneizar la respuesta.

Por otra parte, cada espectro está dividido en grupos de 1024 llamados órdenes, pues de tal forma los registra el espectrógrafo ELODIE. Se eliminó la modulación del espectro producida por el perfil de brillo en cada orden del espectro. Para ello se ajustó el continuo de cuatro estrellas de muy baja metalicidad, a través de polinomios de entre 7° a 19° grado, y se eligió el mejor polinomio de los cuatro para cada orden. Después, para el resto de los espectros, se dividió cada valor de flujo en cada orden por el valor correspondiente del polinomio escogido en dicha longitud de onda. El efecto de esta transformación es que los espectros se “estiran”, la mayoría de los píxeles, exceptuando las líneas de absorción, aparecen en una línea horizontal.

También se eliminaron los órdenes de longitud de onda más bajo por estar mucho más degradados.

Eliminación de rayos cósmicos y píxeles defectuosos

En el intervalo de temperatura y longitudes de onda considerado, solo se pueden observar líneas de absorción. Por lo tanto cualquier píxel (es decir, valor de flujo) que sobrepase el continuo en un incremento determinado puede ser eliminado.

En un segundo paso, se calculó el polinomio de segundo grado ajustado a cada cuatro píxeles consecutivos. Aquellos píxeles cuyos valores residuales presentan una alta dispersión respecto al promedio pueden ser considerados como píxeles defectuosos o impactados por un rayo cósmico, y también pueden ser eliminados.

El tercer método consiste en distinguir los píxeles que, como consecuencia del instrumento, siempre presentan valores defectuosos, y que también fueron eliminados.

En un cuarto paso, se obtiene el espectro del cielo (no de la estrella), y se transforma dicho espectro para eliminar rayos cósmicos, píxeles defectuosos y restarle el continuo. Después, se utiliza el espectro del cielo para restárselo, píxel por píxel, al espectro de la estrella.

Por último, se eliminan las líneas telúricas, cuyas longitudes de onda son conocidas de antemano.

2.1.2 Comparación del espectro con el conjunto de referencia

La forma de comparar dos espectros se fundamenta en el valor de χ^2 reducido entre el espectro estudiado y el espectro de referencia. Para ello, se realizan los siguientes pasos:

- 1) Adecuación de las longitudes de onda. Debido a la diferencia que pueda haber entre las velocidades de las dos estrellas a comparar, se realiza un desplazamiento en el espectro de la estrella de referencia para hacer coincidir exactamente las líneas de absorción de ambos espectros. Además, es necesario hacer una interpolación ya que los valores de las longitudes de onda desplazados pueden no coincidir exactamente.
- 2) Ajuste de los valores de flujo. Los valores de flujo entre las estrellas también varían mucho, por lo que es necesario ajustarlos en una escala común, moviendo el espectro de referencia al nivel medio del espectro. Para ello, se ajusta cada orden del espectro de referencia buscando el factor que aplicado a cada píxel, haga mínima la suma de las diferencias al cuadrado de cada espectro.
- 3) Ponderación de cada orden en función de su relación señal-ruido (*signal-to-noise*). Los espectros de ELODIE están divididos en órdenes, y cada uno de ellos ocupa un rango

determinado de longitudes de onda. En las comparaciones entre los espectros, se valoran más las que más relación señal-ruido tengan.

- 4) Ponderación de cada orden en función de la cantidad de información que contienen. Algunos órdenes contienen varias líneas de absorción mientras que otros carecen de ellas. Para cada estrella de referencia se seleccionaron los 15 órdenes que más información contenían. Para ello, se calculó para cada orden la desviación de la media cuadrática con respecto a la media aritmética del flujo, estimando que los valores más altos correspondían a órdenes más significativos.
- 5) Comparación final. Finalmente, se compara la estrella a estudiar con cada una de las estrellas de referencia. Las 10 estrellas que presentan menor valor de χ^2 reducido son seleccionadas para calcular el valor final, que es la media ponderada de las estrellas cuyo χ^2 reducido es menor que el 111%

2.1.3 Resultados

Para evaluar el sistema, se estimaron los parámetros físicos de cada estrella de referencia con respecto al resto. Este tipo de evaluación se conoce como *leave-one-out*. La desviación típica del error en los diferentes parámetros fue de 86 K, 0.28 dex, y 0.16 dex para la temperatura efectiva, el logaritmo de la gravedad y la relación [Fe/H], respectivamente. El sesgo fue de 17K, 0.04 dex y 0.02 dex para los mismos parámetros, respectivamente.

2.2 El algoritmo MATISSE

La estrategia del algoritmo MATISSE(2), al igual que en el artículo anterior, está fundamentada en el cálculo las distancias de la estrella objetivo a un conjunto de modelos de estrellas, y la interpolación de los parámetros deseados a partir de los parámetros de las estrellas vecinas.

El algoritmo está inspirado en el Análisis de Componentes Principales (PCA). En PCA, se calcula una combinación lineal del espectro cuyas primeras componentes acumulan la mayor parte de la varianza del espectro original. Los pesos de la combinación lineal son los vectores propios (*eigenvectors*) de la matriz de covarianza. Sin embargo, MATISSE trata de calcular los pesos que producen coeficientes lo más cercanos posibles al parámetro físico buscado. Para ello, se busca la correlación estadística entre la entrada (el espectro) y la salida (el parámetro físico). El resultado de esta búsqueda es un vector B_θ por cada parámetro físico θ (temperatura efectiva, gravedad...). La proyección de un determinado espectro a través de este vector da como resultado la estimación del parámetro físico.

El proceso es como sigue: Inicialmente, se restan las respectivas medias tanto a los parámetros físicos como a los espectros. El vector base $B_\theta(\lambda)$ del parámetro θ se define como:

$$B_\theta(\lambda) = \sum_i \alpha_i S_i(\lambda)$$

donde λ es el conjunto de longitudes de onda, que se encuentran en el rango [8475, 8745] Å

$S_i(\lambda)$ es el valor i -ésimo del espectro dentro del conjunto de longitudes de onda considerado

α_i el peso i -ésimo asociado al espectro $S_i(\lambda)$

La estimación del parámetro θ_i para un espectro $S(\lambda)$, denotada como $\hat{\theta}_i$, es la proyección de ese espectro sobre la base B :

$$\hat{\theta}_i = \sum_{\lambda} B_{\theta}(\lambda) S_i(\lambda)$$

Combinando las ecuaciones anteriores:

$$\hat{\theta}_i = \sum_j c_{ij} \alpha_j$$

donde c_{ij} pueden ser interpretados como la covarianza entre los espectros S_i y S_j

α_j son el resultado de resolver la ecuación:

$$\sum_k \left(\sum_i c_{ij} c_{ik} \right) \alpha_k = \sum_i c_{ij} \theta_i$$

En un segundo paso, se vuelve a repetir el método anterior restringiendo el análisis al conjunto de modelos cuyos parámetros se encuentran dentro de un rango cercano al inicialmente predicho. Este segundo mejora enormemente las predicciones debido a la falta de linealidad a escalas locales en los distintos parámetros.

El conjunto de entrenamiento contenía modelos con temperaturas efectivas entre 4000 K y 8000 K, gravedades logarítmicas entre -1 y 5, y metalicidades entre -5 y 1. Para un conjunto de validación de espectros sintéticos con ratio señal-ruido de 50, los errores medios máximos fueron de entre 50K y 150K para la temperatura efectiva, entre 0,04 y 0,2 en el logaritmo de la gravedad y de entre 0,04 y 0,3 para la metalicidad, dependiendo del tipo de estrella considerada: enanas frías de alta metalicidad, gigantes frías de metalicidad intermedia, y subgigantes calientes de baja metalicidad. El artículo no ofrece resultados numéricos del error sobre los anteriores subconjuntos de estrellas.

En un segundo experimento, se vuelve a aplicar el mismo algoritmo sobre los espectros a los que se ha sustraído el continuo. Los resultados fueron muy parecidos, y solo se observó un leve incremento del error en la predicción de la temperatura, y en logaritmo de la gravedad en los dos primeros grupos de estrellas.

El estudio también contempla estrellas reales en sus pruebas: el Sol, cuyos errores fueron de 19 K, 0,11 dex y de 0,07 para la temperatura efectiva, logaritmo de la gravedad y metalicidad, y la estrella Arcturus, con errores de 23K, 0,2 dex y 0,1 dex en los parámetros anteriores.

2.3 El método “piecewise PLS”

En (5) se propone la utilización de la regresión PLS (*Partial Least Squares*) para la estimación de los parámetros físicos estelares. El objetivo de PLS es encontrar componentes de las variables de entrada (X) que sean también relevantes para las variables de salida (Y), buscando una descomposición de X e Y que explique la covarianza entre ambos conjuntos tanto como se pueda. En un segundo paso de regresión, la descomposición de X es utilizada para predecir Y.

La forma del continuo está principalmente determinada por la temperatura efectiva, mientras que la combinación del continuo y las líneas de absorción o emisión determinan todos los parámetros físicos (gravedad, metalicidad y temperatura). Para separar mejor la influencia de cada parámetro en el espectro, el artículo propone estimar primero la temperatura mediante regresión por PLS, y en función de la temperatura estimada, utilizar un modelo regresivo u otro (de un conjunto de 7 posibles) que ajuste mejor el logaritmo de la gravedad y la metalicidad. Los 7 conjuntos cubren el siguiente rango de temperaturas: O, >25000 K; B: 11000 K – 25000 K; A: 7500 K – 11000 K; F: 6000 – 75000 K; G: 5000 K – 6000 K; K: 3500 K – 5000 K; y M: < 3500 K. A esta variación del algoritmo, los autores la denominan *piecewise PLS*.

La medida de error que se utiliza en el artículo es el sesgo ($\mu = \frac{1}{n} \sum_{i=1}^n error_i$) y la desviación

típica respecto al sesgo ($\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (error_i - \mu)^2}$). En un primer experimento, se utilizó la mitad de la librería MILES (6) como conjunto de entrenamiento y la otra como conjunto de test, y los resultados fueron de $\mu=0.50\%$ y $\sigma=6.66\%$ para T_{eff} , y de $\mu=0.0006$ dex y $\sigma=0.5463$ dex para $\log(g)$. En posteriores experimentos, se incluyen variantes del anterior utilizando espectros degradados introduciendo ruido gaussiano y reduciendo la resolución.

En otro de los experimentos, se utilizó como conjunto de test una selección de 700 estrellas de ELODIE versión 3(11). Esta versión de la colección de ELODIE es diferente a la que se hizo referencia en el apartado 2.1. Las estrellas seleccionadas tenían una temperatura efectiva de entre 5000K y 15000K con el indicador de calidad mayor o igual a 1. Los resultados fueron de $\mu=-3.78\%$ y $\sigma=5.20\%$ para T_{eff} , de $\mu=0.1167$ dex y $\sigma=0.4864$ dex para $\log(g)$, y de -0.1285 y 0.4147 para la metalicidad.

3 Solución propuesta

La investigación llevada a cabo en este trabajo parte de tres fuentes principales de datos: los modelos de Kurucz, los espectros reales del espectrógrafo FLAMES, y la selección de estrellas clasificadas del espectrógrafo ELODIE versión 3.1 (11).

Los modelos de Kurucz son un conjunto de espectros sintéticos creados por Robert L. Kurucz. Son modelos matemáticos utilizados para determinar los espectros (diferentes valores de brillo en diferentes longitudes de onda del espectro electromagnético), a partir de la metalicidad¹, velocidad turbulenta, temperatura efectiva y gravedad de una supuesta estrella. Estos modelos relacionan un espectro con los distintos parámetros físicos de la estrella, y constituyen el conjunto de evidencias a partir del cual habrá que construir un modelo para la predicción de los parámetros físicos de nuevas estrellas.

La segunda fuente de datos son un conjunto de espectros de estrellas reales procedentes del espectrógrafo FLAMES (Fibre Large Array Multi Element Spectrograph) del telescopio VLT (*Very Large Telescope*) de la ESO (European Southern Observatory), instalado en Cerro Paranal, Chile. El espectrógrafo FLAMES (3) puede registrar el espectro en el rango visible (3700-9500Å) con una resolución entre $R \sim 10000$ y $R \sim 47000$, lo que viene a representar una separación entre longitudes de onda de entre 0,037 y 0,95 Å, dependiendo de la longitud de onda y resolución consideradas. Estos espectros no tienen asignado ningún parámetro físico. En el transcurso de los trabajos, fueron proporcionados dos conjuntos de espectros procedentes de FLAMES, que serán referidos como FLAMES I y FLAMES II a lo largo del presente documento.

La idea original de la investigación fue utilizar los espectros sintéticos de los modelos de Kurucz para estimar los parámetros físicos de los espectros de FLAMES. Sin embargo, la ausencia de estimaciones de los parámetros físicos de los espectros de FLAMES hacía difícil determinar la precisión de los modelos predictivos. Obviamente se podrían evaluar los modelos utilizando la propia librería de Kurucz y técnicas estándar de Minería de Datos, como la validación cruzada en cualquiera de sus variantes. Sin embargo, la naturaleza teórica de los modelos utilizados en el entrenamiento conlleva diferencias sistemáticas no soslayables respecto a los espectros observados, lo que hacía aconsejable complementar las medidas de precisión intrínsecas con otros análisis

1 Proporción de la materia de una estrella exceptuando hidrógeno y helio, respecto su masa total.

extrínsecos (llevados a cabo sobre espectros observados). Esto llevó a utilizar un tercer conjunto de espectros reales que estuvieran ya clasificados, los espectros clasificados del espectrógrafo ELODIE en su versión 3.1(11). Este conjunto de datos está formado por 1968 espectros entre los 3900 y 6800 Å en dos resoluciones diferentes: $R=42.000$ y $R=10.000$. Se escogió la última por coincidir con la resolución que se tenía para FLAMES. Para cada espectro se había estimado su temperatura efectiva (3000K a 60000 K), el logaritmo de la gravedad (-0,3 a 5,9) y el índice de metalicidad ([Fe/H]). Sobre este conjunto de espectros, se seleccionó aquéllos cuyo indicador de calidad era mayor o igual a 1. Este mismo filtro se realizó en el algoritmo MATTISE, véase el apartado 2.2.

Existen varias diferencias entre los conjuntos de datos, lo que hizo necesario una armonización de ambos conjuntos que se realizó en la fase de preparación de datos. Esas diferencias son:

- Falta de coincidencia en las longitudes de onda entre los distintos conjuntos de datos.
- Diferentes valores para el flujo medio de las estrellas o modelos de estrellas.
- Ruido en los espectros de estrellas reales, FLAMES y ELODIE.

Otros aspectos relacionados con la preparación de los datos son:

- La existencia de líneas de absorción telúricas, producidas por la interferencia de la atmósfera terrestre en el espectro estelar, en los espectros de estrellas reales.
- El desplazamiento de los espectros FLAMES debido al efecto doppler.
- Un gran número de variables a considerar. Hay que tener en cuenta que cada espectro puede contener entre 2000 y 3000 valores de flujo, tantos como longitudes de onda hayan sido consideradas. Este número es intratable en algunas técnicas de minería de datos, por lo que se hace necesario reducir el número de valores de flujo de los espectros, evitando limitar la información que contienen, a través de técnicas de reducción de la dimensionalidad.
- Sustracción del espectro continuo. El análisis de un espectro se realiza con mayor precisión si se elimina el espectro continuo, que es el producido por la estrella de acuerdo a una curva de distribución dependiente únicamente de su temperatura y de cada longitud de onda considerada.

Las soluciones adoptadas para los problemas anteriores se expondrán en el apartado 3.2, de preparación de los datos.

Una vez reducido el número de variables a considerar, se aplicaron técnicas de minería de datos para construir un modelo de regresión para los parámetros elegidos. Estos modelos fueron los que se aplicaron sobre los espectros reales (ELODIE y FLAMES) para estimar su bondad.

3.1 Entorno y herramientas de desarrollo

Las características del equipo utilizado para el desarrollo y ejecución de los experimentos fueron:

- Sistema operativo: Ubuntu 11.04
- Procesador: Intel Core Duo 1.66GHz
- Memoria 2GB

Las herramientas utilizadas para el tratamiento de datos fueron fundamentalmente R ⁽²⁾, y en menor

² <http://www.r-project.org/>

medida *Weka* (³). La herramienta clásica *make* fue utilizada para ejecutar de forma ordenada los programas, mientras que *awk* se utilizó para transformar los datos de las fuentes originales en un formato entendible por *R* o *Weka*, o para transformar datos entre estos dos programas. Todas las gráficas fueron generadas con *R*. Este documento fue redactado con *LibreOffice*. En total, se escribieron unas 8.000 líneas de código en *R*.

3.2 Preparación de los datos

En los siguientes apartados se irán describiendo el conjunto de transformaciones que sufren los espectros antes de pasar a la fase de minería de datos.

3.2.1 Interpolación

Los modelos de Kurucz y los espectros FLAMES se encuentran en formatos diferentes, y fue necesario adaptar uno al otro para que estén referidos a las mismas longitudes de onda.

En el caso de los modelos de Kurucz, los espectros se encuentran en un directorio que contiene:

- El fichero *wave.dat*, que contiene una sola columna con las longitudes de onda en Angstroms.
- Varios ficheros de tipo *spectrum.**, que contienen el flujo correspondiente a cada longitud de onda del fichero anterior en la primera columna.

Los datos de FLAMES se encuentran distribuidos en niveles: el primer nivel son los campos o regiones del cielo a las que pertenecen las estrellas observadas, y el segundo es el rango en longitudes de onda del espectro (LR2 o LR6; los valores precisos de los rangos se proporcionan más abajo). Los ficheros de extensión *.dat* en estos directorios son de tipo texto, y contienen una lista de dos columnas; la primera con las longitudes de onda y la segunda con el flujo. Cada fichero corresponde a una estrella diferente.

Las longitudes de onda dentro de los ficheros *.dat* en el directorio *LR2* están en el rango de 396 a los 457,1 nanómetros, y la de los ficheros *.dat* en *LR6* entre los 643,7 a los 718,28 nanómetros. Ambos ficheros suelen contener ceros en el flujo al final del fichero. Nótese que existe una laguna de información entre los 457,1 y los 643,7 nanómetros.

La Tabla 1 muestra una comparación de diversos valores estadísticos y otros datos entre las longitudes de onda de los modelos de Kurucz y los de FLAMES.

Los valores de longitud de onda de los espectros de FLAMES fueron convertidos a angstroms. Para los valores de flujo, fue necesario llevar a cabo una interpolación haciendo coincidir los valores de flujo de los modelos de Kurucz (el espectro más denso⁴) con los valores de los datos en FLAMES (el espectro menos denso). Inicialmente, la interpolación se realizó en dos rangos de longitudes de onda: de 3960 a 4571 y de 6437 a 7000 angstroms, que son los dos rangos en los que se solapan las dos fuentes de datos. Por tanto, los datos en los rangos 3500-3959 Å, 4571-6437 Å, y 7001 – 7182,8 Å fueron ignorados. A partir de ahora, al primero de los rangos se lo denominará *rango azul*, y al segundo *rango rojo*, por su correspondencia con los colores en el espectro visible.

³ <http://www.cs.waikato.ac.nz/ml/weka/>

⁴ Entiéndase por densidad del espectro como el número medio de valores de flujo en un rango fijo de longitudes de onda.

	Modelos de Kurucz	Espectros de FLAMES	Comentarios
Unidades de la longitud de onda	Angstroms	Nanometros	En adelante, los datos referidos a FLAMES se mostrarán en angstroms.
Rango	3500 - 7000	3960 - 4571 y 6437 - 7182,8	En angstroms. Nótese que existe una <i>laguna</i> en los datos FLAMES, en el rango 4571-6437A
Diferencias entre longitudes de onda contiguas	0,0875 - 0,175 Media: 0,12	0,2	En angstroms. Nótese que la diferencia entre longitudes de onda de los modelos de Kurucz no es constante.
Número medio valores por cada 1000 Angstroms	7923	5002 en ambos rangos	No se tiene en cuenta la laguna de datos de FLAMES, en el rango 4571-6437

Tabla 1: Diferencias entre los conjuntos de datos de Kurucz y de FLAMES I

A raíz de la incorporación del conjunto de datos de ELODIE, se decidió reducir el rango rojo a 6437 – 6800Å. Con la incorporación del conjunto de espectros de FLAMES II, se descubrieron espectros que no se ajustaban exactamente a dichos rangos, por lo que se volvieron a modificar ligeramente los límites, quedando definitivamente establecidos en:

	Longitud de onda inicial	Longitud de onda final
Rango azul	3960,2	4564
Rango rojo	6439,4	6799

Se estableció que la resolución del espectro fuera la de un valor de flujo por cada 0.2 Å, como aparecen en FLAMES. De esta forma, se iban a tener 1799 y 3020 valores de flujo por cada espectro, para el rango azul y rojo respectivamente.

El problema de la interpolación consiste en encontrar un valor de flujo (f) para la longitud de onda (w). Existen varios tipos de interpolación, y la elegida fue la interpolación polinómica de Newton de diferencias divididas. Para definir esta interpolación hay que definir previamente las diferencias divididas como sigue:

$$f[w_i, w_j] = \frac{f(w_i) - f(w_j)}{w_j - w_i}$$

$$f[w_i, w_j, w_k] = \frac{f[w_i, w_j] - f[w_j, w_k]}{w_i - w_k}$$

$$\vdots$$

$$f[w_n, w_{n-1}, \dots, w_1, w_0] = \frac{f[w_n, w_{n-1}, \dots, w_2, w_1] - f[w_{n-1}, w_{n-2}, \dots, w_1, w_0]}{w_n - w_0}$$

donde w_x representa una longitud de onda tomada del modelo de Kurucz, y $f(w_x)$ representa el flujo para la longitud de onda w_x en el modelo de Kurucz.

A partir de las diferencias divididas, la interpolación polinómica de Newton se define como:

$$\begin{aligned}
 F_n(w, w_0, w_1, \dots, w_n) = & \\
 & f(w_0) + \\
 & + f[w_1, w_0](w - w_0) + \\
 & + f[w_2, w_1, w_0](w - w_1)(w - w_2) + \\
 & \vdots \\
 & + f[w_n, w_{n-1}, \dots, w_2, w_1, w_0](w - w_0)(w - w_1) \cdots (w - w_{n-1})
 \end{aligned}$$

De esta forma, la interpolación lineal se realizaría con dos puntos:

$$F_1(w, w_0, w_1) = f(w_0) + \frac{f(w_1) - f(w_0)}{w_1 - w_0} (w - w_0)$$

donde

$F_1(w, w_0, w_1)$ es el valor de *flux* para la nueva longitud de onda w
 w_0, w_1 son las longitudes de onda inmediatamente inferior y superior del modelo de Kurucz
 $f(w_1)$ es el valor de *flux* para w_1 en el modelo de Kurucz
 $f(w_0)$ es el valor de *flux* para w_0 en el modelo de Kurucz

Por ejemplo, para la interpolación cuadrática, se utilizarían los tres puntos del modelo de Kurucz más cercanos al deseado. La interpolación desarrollada quedaría:

$$\begin{aligned}
 F_2(w, w_0, w_1, w_2) = & f(w_0) + \frac{f(w_1) - f(w_0)}{w_1 - w_0} (w - w_0) + \\
 & + \frac{\frac{f(w_2) - f(w_1)}{w_2 - w_1} - \frac{f(w_1) - f(w_0)}{w_1 - w_0}}{w_2 - w_0} (w - w_0)(w - w_1)
 \end{aligned}$$

donde

$F_2(w, w_0, w_1, w_2)$ es el valor de *flux* para la nueva longitud de onda w
 w_0, w_1, w_2 son las longitudes de onda más cercanas a w en el modelo de Kurucz,
ordenadas ascendentemente
 $f(w_0), f(w_1), f(w_2)$ son los *flux* para w_0, w_1, w_2 , respectivamente, en el modelo de Kurucz

Fue la la interpolación cúbica la utilizada para los propósitos buscados, lo que supone buscar los cuatro puntos más cercanos, dos delante y dos detrás. Se muestra la fórmula sin desarrollar:

$$\begin{aligned}
 F_4(w, w_0, w_1, w_2, w_3) = & \\
 & f(w_0) + f[w_1, w_0](w - w_0) + f[w_2, w_1, w_0](w - w_1)(w - w_2) + \\
 & + f[w_3, w_2, w_1, w_0](w - w_0)(w - w_1)(w - w_2)(w - w_3)
 \end{aligned}$$

La Ilustración 1 muestra un ejemplo de los flujos originales (círculos azules) e interpolados (puntos rojos) en un modelo de Kurucz, para un rango de unos 200 valores de flujo.

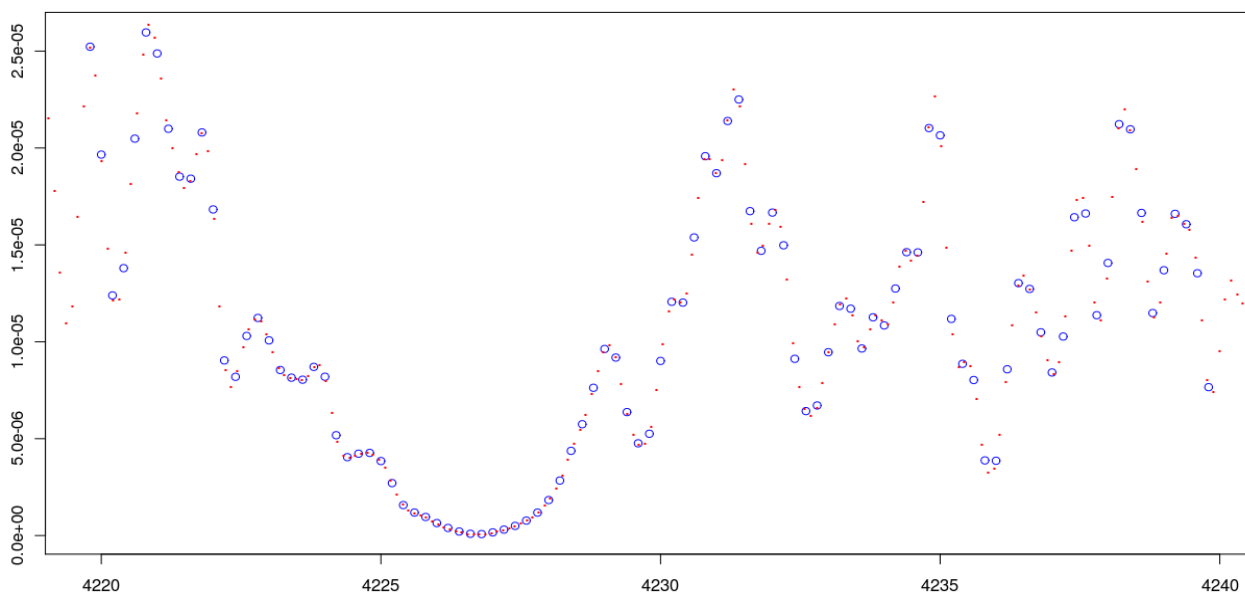


Ilustración 1: Ejemplo de los puntos originales e interpolados de flujo en un modelo de Kurucz

3.2.2 Eliminación del ruido

Los espectros FLAMES y de ELODIE contienen algo de lo que carecen los modelos de Kurucz: ruido. Es necesario eliminar las componentes de alta frecuencia del espectro pues esto permitirá tener mayor confianza en que la comparación entre los espectros reales (FLAMES o ELODIE) y los modelos de Kurucz sea más precisa.

Para tal propósito, se utilizaron las transformadas *wavelet* de los espectros. Una definición simple de la transformada *wavelet* de un espectro podría ser: una colección de representaciones de frecuencia a distintas escalas. La aplicación de la transformada *wavelet* sobre una señal (en nuestro caso, el espectro) devuelve como resultado una serie de coeficientes *wavelet*.

Matemáticamente, esta transformación puede expresarse con un producto de matrices, donde W es la matriz de transformación *wavelet*, s es el espectro y w son los coeficientes *wavelet*:

$$w = Ws$$

Además, la transformada *wavelet* es invertible, es decir, a partir de los coeficientes se puede volver a calcular la señal original: $s = W^T w$.

Los coeficientes *wavelet* están estructurados en niveles según la escala de frecuencia. En la práctica, esto significa que se dividen en grupos donde los primeros niveles contienen la información de mayor frecuencia, y en los siguientes se va disminuyendo la frecuencia de la información representada. Por este motivo, se suele hacer referencia a los coeficientes *wavelet* individuales como $W_{j,k}$, donde j es el nivel de resolución y k el coeficiente individual dentro de ese nivel de resolución.

En (12) se demuestra cómo la transformada *wavelet* puede ser una buena herramienta de supresión de ruido. En el artículo se hallan los coeficientes *wavelet* de varias señales ficticias con y sin ruido blanco. Los coeficientes de señales sin ruido concentran sus valores más altos en un pequeño subconjunto, mientras que el resto es cero o casi cero. En cambio, los coeficientes *wavelet* de señales con ruido quedan afectados de una forma muy particular, ya que el ruido contamina por igual a todos ellos, aunque mantienen los valores altos producidos por la señal. Esta circunstancia

motivó la idea de que se podría reconstruir la señal utilizando solo un subconjunto de los coeficientes wavelet, aquellos que contribuían a la reconstrucción de la señal, y anulando el resto de coeficientes. La Ilustración 2 muestra un esquema del proceso.

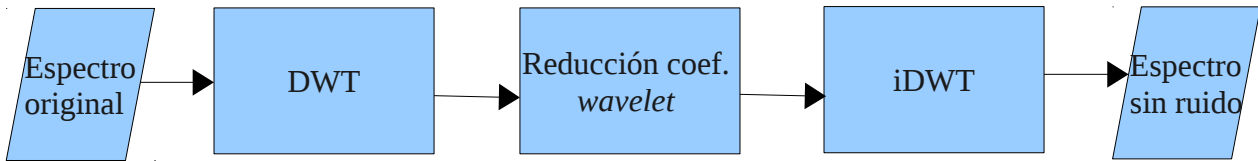


Ilustración 2: Proceso de eliminación del ruido a través de wavelets

Dado que los componentes *wavelet* que contribuyen a la señal tienen valores altos y los que provienen del ruido tienen valores bajos, la forma más natural de seleccionar los coeficientes *wavelet* de la señal fue la de establecer un límite por debajo del cual los coeficientes se anularan, y por encima se mantuviesen con sus valores originales. La forma elegida para escoger el valor de este límite fue:

$$\delta = \sqrt{2 \sigma^2(\text{ruido}) \log(N)}$$

donde δ es el límite

$\sigma^2(\text{ruido})$ es la varianza estimada del ruido, calculada como la desviación absoluta respecto a la mediana en cada uno de los niveles de resolución de los coeficientes *wavelet*

N es el número de valores de la señal original.

Este mecanismo de eliminación de ruido puede ser resultar insuficiente en algunos casos. Por ese motivo, existe otra forma de eliminación de ruido denominada *soft* (en contraposición con la anterior, llamada *hard*) cuya estrategia consiste en anular los coeficientes por debajo del límite establecido y reducir el resto en ese mismo límite.

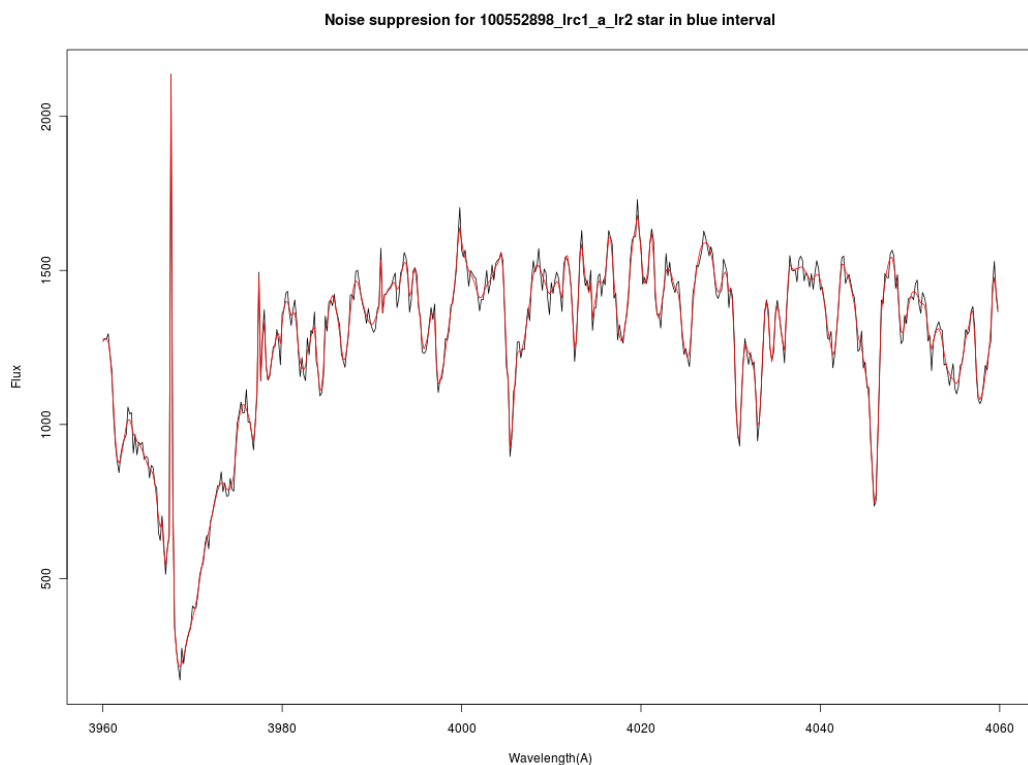
$$W_s = \begin{cases} 0 & \text{si } W_{j,k} < \delta \\ W_{j,k} - \delta & \text{en otro caso} \end{cases}$$

donde W_s es el conjunto de componentes wavelet después de la reducción.

En un punto intermedio entre las dos formas anteriores de reducción de ruido se encuentra un mecanismo conocido como *mid*, que anula los coeficientes por debajo un límite, los reduce utilizando la estrategia *soft* entre δ y 2δ , y no modifica el valor a partir de 2δ :

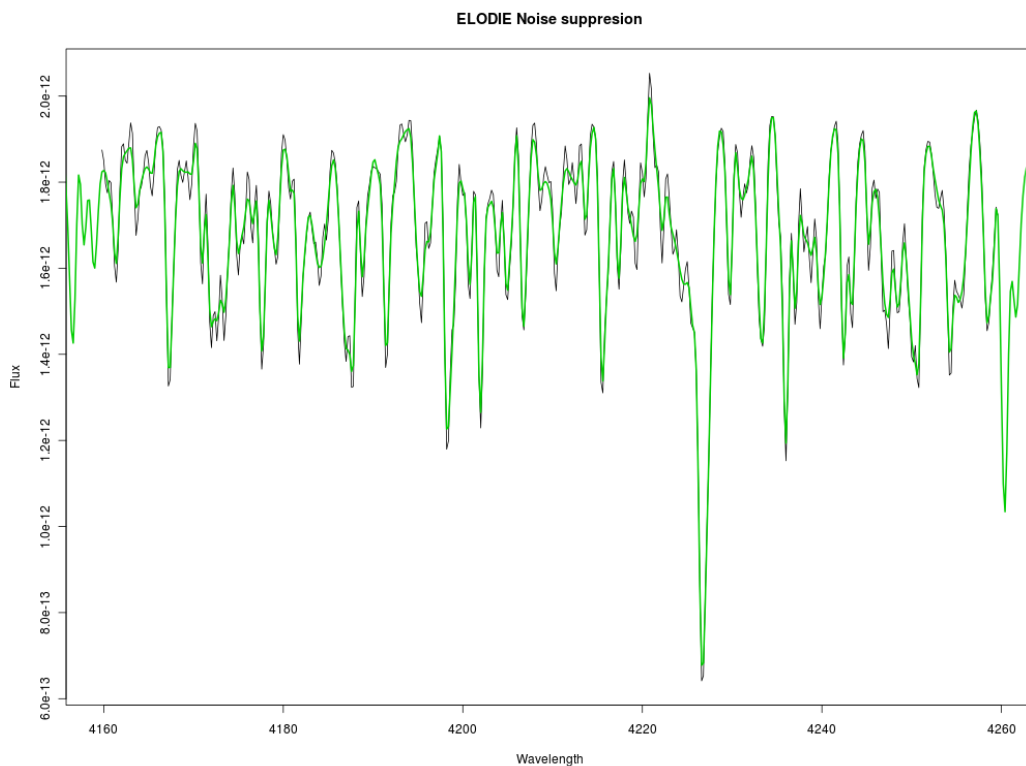
$$W_s = \begin{cases} 0 & \text{si } W_{j,k} < \delta \\ W_{j,k} - \delta & \text{si } \delta < W_{j,k} < 2\delta \\ W_{j,k} & \text{en otro caso} \end{cases}$$

La forma de elección del mejor reductor de ruido fue por inspección del resultado. Entre las tres formas anteriores, se estimó que el mecanismo *mid* era el que mejor resultados ofrecía. Existen otras formas de ajustar el algoritmo de reducción de ruido, como la modificación de δ a través de un factor constante, u otras formas alternativas de su cálculo, pero no se consideró necesario explorarlas. La transformación se llevó a cabo a través de la función *wavShrink* del paquete *wmtsa* de R. Esta función permite la reducción de ruido utilizando el método descrito anteriormente.



*Ilustración 3:
Efecto de la
supresión del
ruido en la
estrella
100552898 de
FLAMES I.*

Se muestra un
extracto del
rango azul de
100 angstroms.



*Ilustración 4:
Efecto de la
supresión del
ruido en la
estrella 00001
de ELODIE.*

Se muestra un
extracto del
rango azul de
100 angstroms.

En la Ilustración 3 se muestra⁵ un extracto en el rango azul de un espectro FLAMES original (en negro) y después de la supresión de ruido (en rojo). Nótese la presencia de un impacto por rayo cósmico (parecido a una línea de emisión estrecha) a la izquierda del espectro original, que no consigue eliminar el filtro de ruido. Este tipo de anomalías se tratan en el apartado 3.2.4. En la Ilustración 4 se muestra un extracto en el rango azul de un espectro ELODIE original (en negro) y después de la supresión de ruido (en verde). Para que se aprecie mejor la reducción de ruido, se muestran solo 100 Å en cada gráfica, que corresponden a 500 píxeles.

3.2.3 Sustracción del continuo

La superficie de un cuerpo sólido muy caliente emite radiación en todas las longitudes de onda, produciendo un espectro que forma una curva que depende únicamente de la temperatura. Esta curva se denomina espectro continuo, o simplemente, continuo, y aparece sumada a los espectros reales, y es conveniente eliminarla del espectro si se desea descubrir información no relacionada con esta característica.

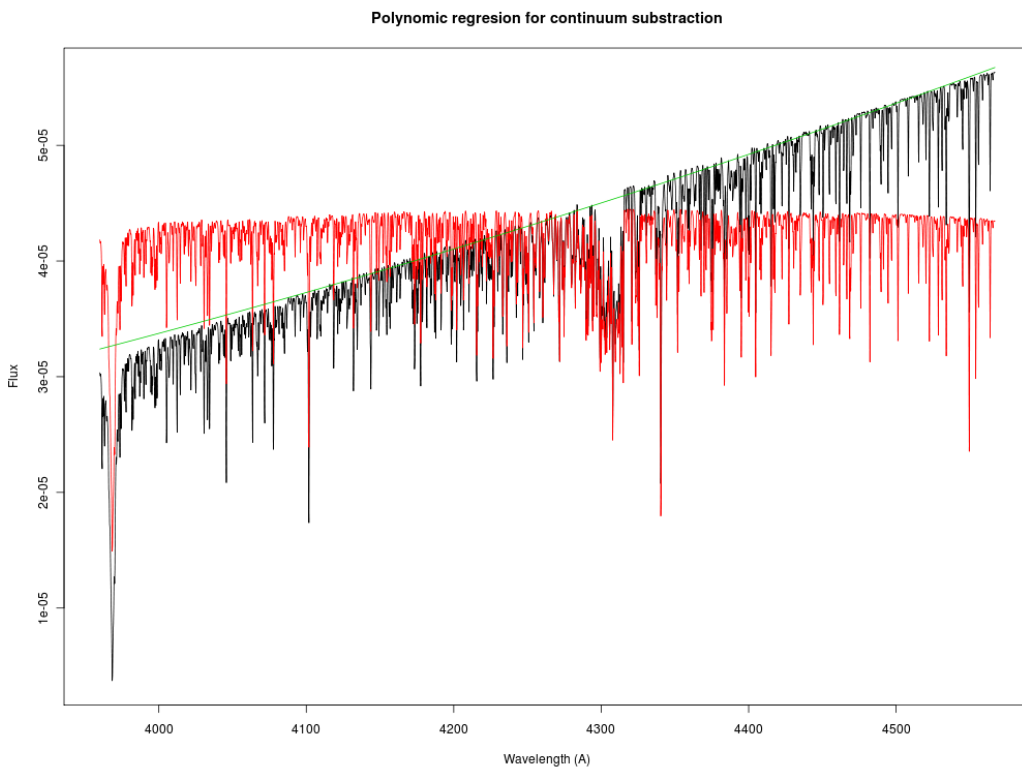
Si se conociera la temperatura de la estrella de antemano, sería posible calcular el espectro continuo, pero no es el caso de esta investigación. En su lugar, se buscó otro método alternativo que sustrajera el continuo del espectro. Para un modelo de Kurucz, el procedimiento es como sigue:

1. Eliminar del espectro las zonas que son más susceptibles de contener profundas líneas de absorción, que son las que más se alejan del continuo. En el rango azul, las zonas eliminadas (expresadas en anstroms) fueron [3960.2, 4010], [4060.2, 4160], [4260.2, 4420], y [4560.2, 4564.0]. En el rango rojo, se eliminó una única zona: [6479.4, 6639.2]
2. Dividir el espectro en 20 intervalos. Eliminar, en cada intervalo, los valores de flujo que se encuentran por debajo del percentil 90, para no incluir valores de flujo que correspondan a líneas de absorción
3. Interpolarse linealmente los valores eliminados a partir de los que han quedado en el espectro.
4. Buscar el polinomio de tercer grado que mejor se ajuste al espectro resultante del punto anterior
5. Restar el polinomio del punto anterior y sumar la media aritmética al espectro original.

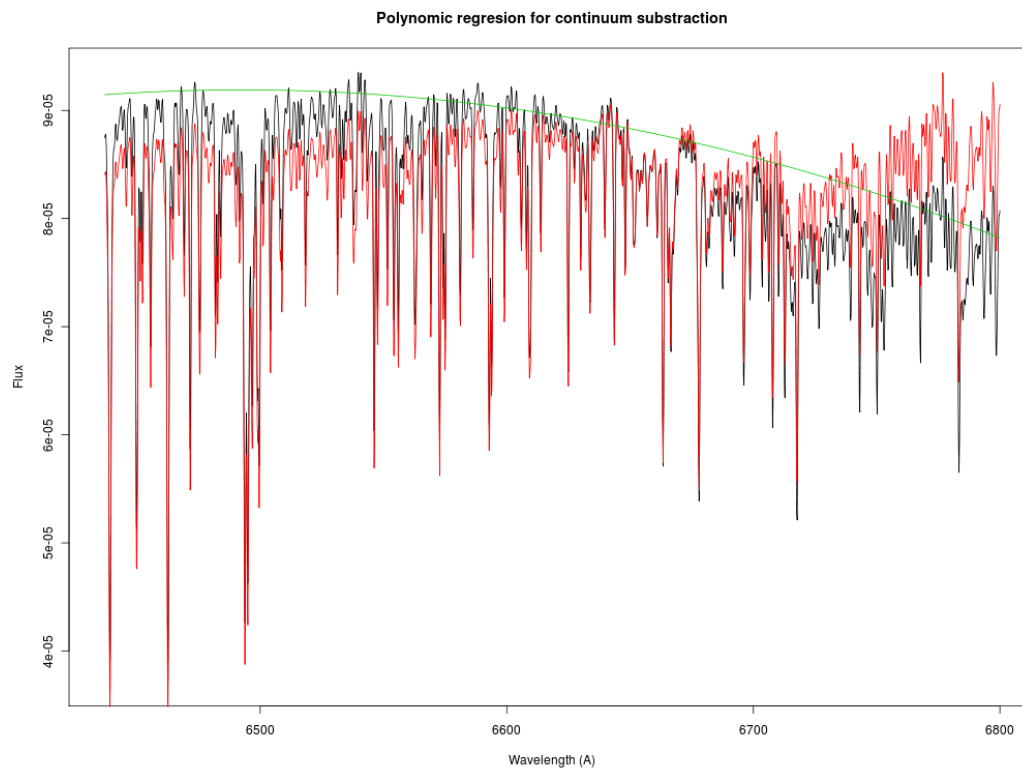
Para un espectro FLAMES, el proceso es similar salvo por el punto 2, donde los flujos eliminados son los que se encuentran por debajo del percentil 80 y por encima del 90. El propósito de esta eliminación es prescindir de las posibles líneas de emisión que pueda contener el espectro, debidas, por ejemplo, a rayos cósmicos.

El efecto de este procedimiento sobre un espectro es el de *aplanar* un espectro, el aspecto que adquiere parece mantener una línea horizontal en su promedio. En la Ilustración 5 se muestra el espectro original (en negro), el espectro después de la sustracción del continuo (en rojo) y el polinomio ajustado (en verde) de un modelo de Kurucz con temperatura efectiva de 4500K, logaritmo de la gravedad 0, para el rango azul. La Ilustración 6 muestra esta misma información en el rango rojo, para un modelo de Kurucz con temperatura efectiva de 4250K y logaritmo de la gravedad 4. La Ilustración 7 y la Ilustración 8 representan la misma información para las estrellas reales de FLAMES, concretamente la estrella 100991167 en el rango azul y la estrella 100919104 en el rango rojo, respectivamente.

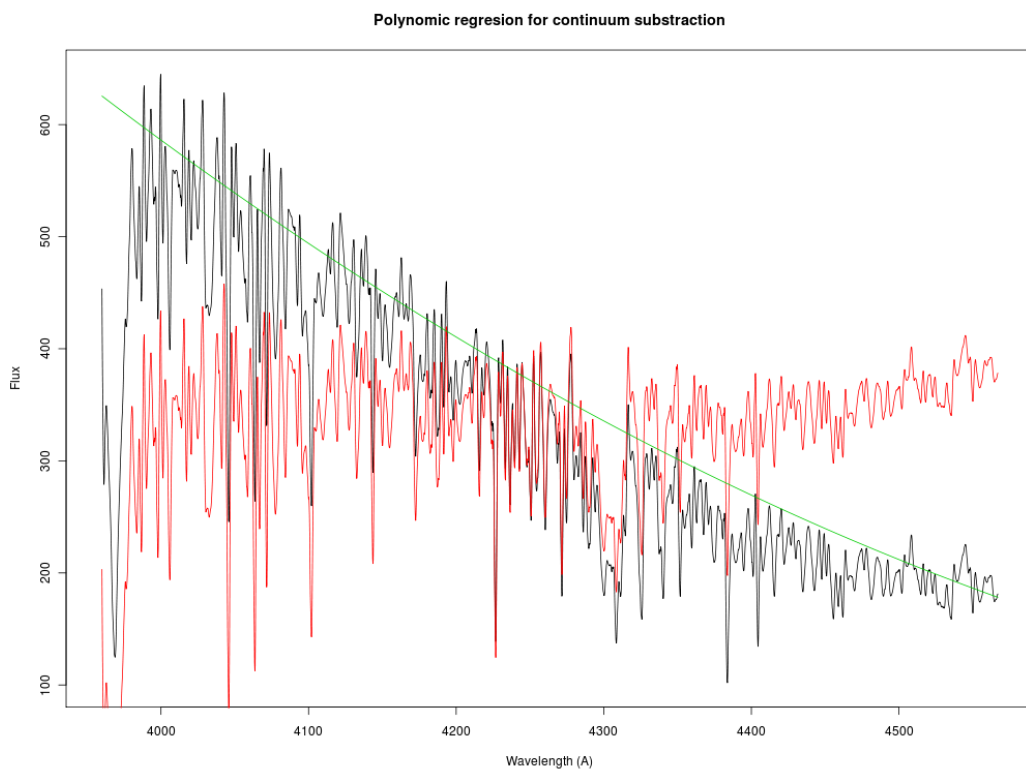
5 El texto de esta gráfica, y de otras en este documento, se encuentra en inglés, con objeto de facilitar su utilización posterior en publicaciones en este idioma.



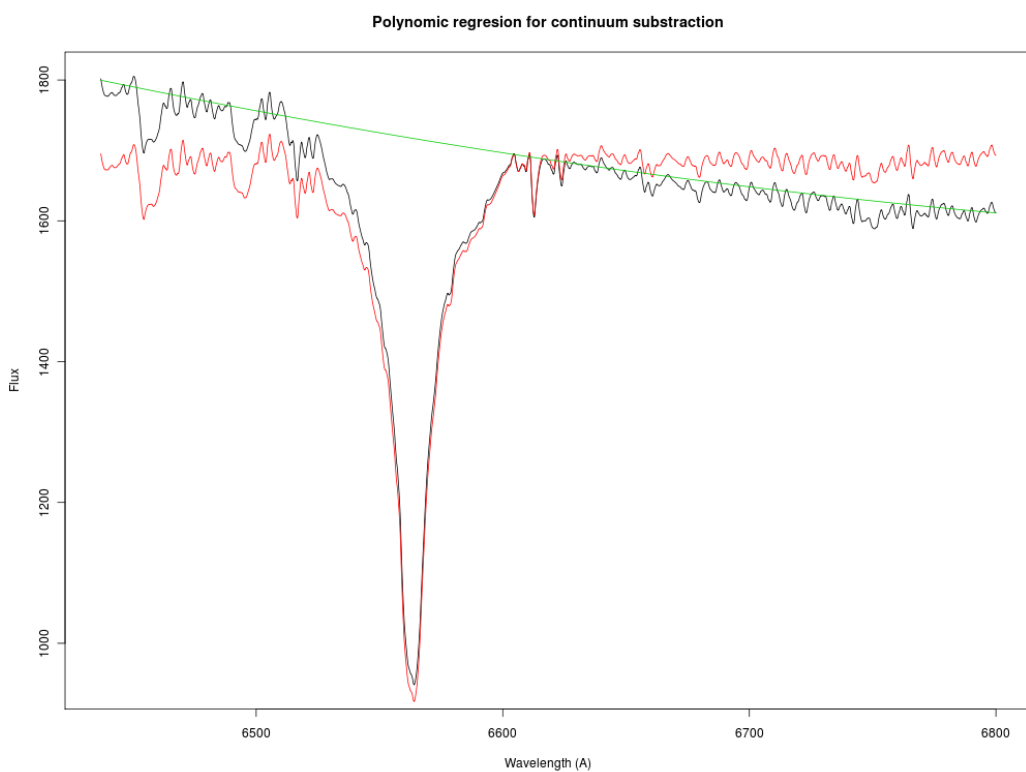
*Ilustración 5:
Sustracción
del continuo
en un modelo
de Kurucz en
el rango azul*



*Ilustración 6:
Sustracción del
continuo de un
modelo de
Kurucz en el
rango rojo*



*Ilustración 7:
Sustracción del
continuo de la
estrella
100991167 de
FLAMES en el
rango azul*



*Ilustración 8:
Sustracción del
continuo en la
estrella
100919104 de
FLAMES en el
rango rojo*

3.2.4 Eliminación de rayos cósmicos

Los rayos cósmicos se producen por el impacto de un fotón de gran energía procedente de otros fenómenos astronómicos, y son completamente ajenos a la naturaleza de la estrella. Por tanto, han de ser considerados como ruido y eliminados. En los conjuntos de datos de FLAMES, aparecieron multitud de espectros con rayos cósmicos que la supresión de ruido no había conseguido eliminar. Por tal motivo, se optó por hacer un filtro específico para la eliminación de rayos cósmicos. Los rayos cósmicos se distinguen por ser valores de flujo de alta energía muy concentrados en no más de 3 o 4 valores de flujo. El algoritmo elegido para su eliminación consiste en la división de cada espectro en conjuntos consecutivos de 100 valores de flujo, y la eliminación dentro de cada conjunto de aquellos que superan el valor de la mediana más 5 veces el valor de la desviación absoluta respecto a la mediana (mad), definida como:

$$mad = \text{mediana}(|x - \text{mediana}(x)|)$$

La razón de escoger los valores de flujo de 100 en 100, es que de esa forma se evita eliminar valores de flujo del espectro que se encuentran muy alejados de la mediana, debidos sobre todo a espectros con componentes de alta frecuencia. Con todo, en algunos espectros todavía pueden anularse valores de flujo que no corresponden a rayos cósmicos, pero se ha comprobado por inspección que estos espectros son muy escasos.

La Ilustración 9 y la Ilustración 10 muestran el efecto del filtro sobre dos espectros de FLAMES I y II, en los rangos azul y rojo, respectivamente. El espectro negro es el original, y el rojo es el que aparece después del filtro. En ambas gráficas, el espectro rojo se superpone encima del negro, salvo en los rayos cósmicos, cuyos valores son interpolados a partir de los valores colindantes.

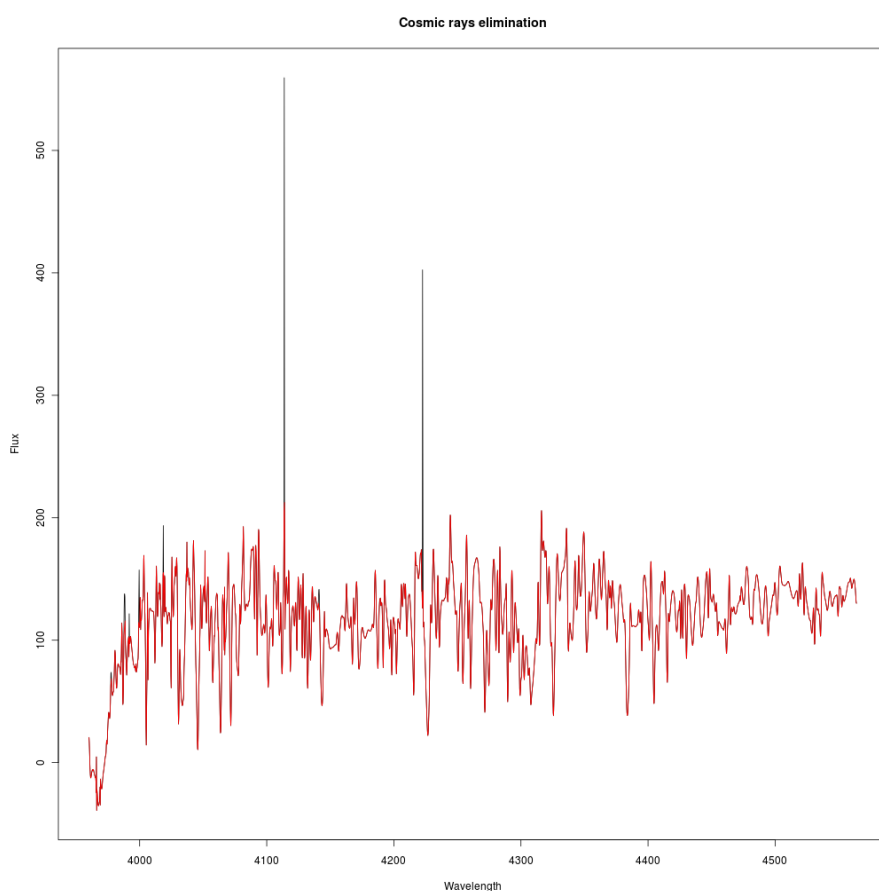


Ilustración 9: Eliminación de rayos cósmicos en la estrella 101091412 (rango azul de FLAMES I).

Las dos líneas negras verticales corresponden a dos rayos cósmicos suprimidos. El espectro después del filtrado se muestra en rojo.

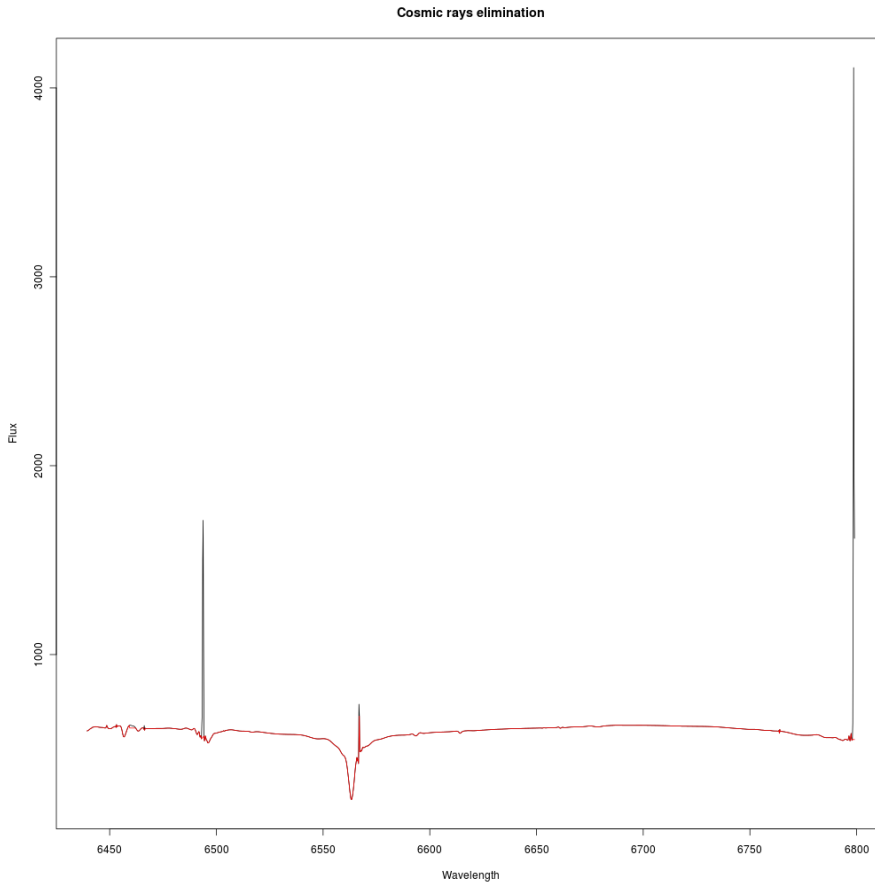


Ilustración 10: Eliminación de rayos cósmicos en la estrella 110827797 (2010-12-13 08h02) Rango rojo, FLAMES II.

Las dos líneas negras verticales corresponden a dos rayos cósmicos suprimidos. Un tercer rayo cósmico, dentro de la línea de absorción, no consigue ser eliminado completamente. El espectro después del filtrado se muestra en rojo.

3.2.5 Estimación de la relación señal-ruido

Los espectros de FLAMES tienen distintas calidades, en algunos de ellos se aprecian de forma mucho más clara las líneas de absorción, mientras que en otros prevalece una componente de alta frecuencia que distorsiona el espectro hasta el punto de hacer indistinguibles la mayoría de líneas de absorción. Para clasificar los espectros en función de la cantidad de ruido que poseen, se pensó utilizar una medida de la relación señal-ruido que consistía en la relación de la desviación típica en una zona del espectro donde apareciera casi siempre una línea de absorción profunda, y otra zona del espectro que careciera (en la mayoría de los casos) de líneas de absorción profundas:

$$snr = \frac{\sqrt{\frac{\sum_{w \in AL} (f(w) - \bar{f}(w))^2}{n_{AL} - 1}}}{\sqrt{\frac{\sum_{w \notin AL} (f(w) - \bar{f}(w))^2}{n_{NAL} - 1}}} = \frac{stdev_{w \in AL}}{stdev_{w \notin AL}}$$

Las zonas elegidas como zonas con y sin líneas de absorción para cada rango de longitudes de onda se expresan en la siguiente tabla:

	Zona con líneas de absorción profunda	Zona sin líneas de absorción profundas
Rango azul	[3962, 4002] Å	[4410, 4450] Å
Rango rojo	[6555.2, 6575.2] Å	[6699.2, 6719.2] Å

Las siguientes ilustraciones muestran diversos espectros y sus medidas señal-ruido. El valor *SNR* concreto del espectro se muestra en el título superior de cada ilustración.

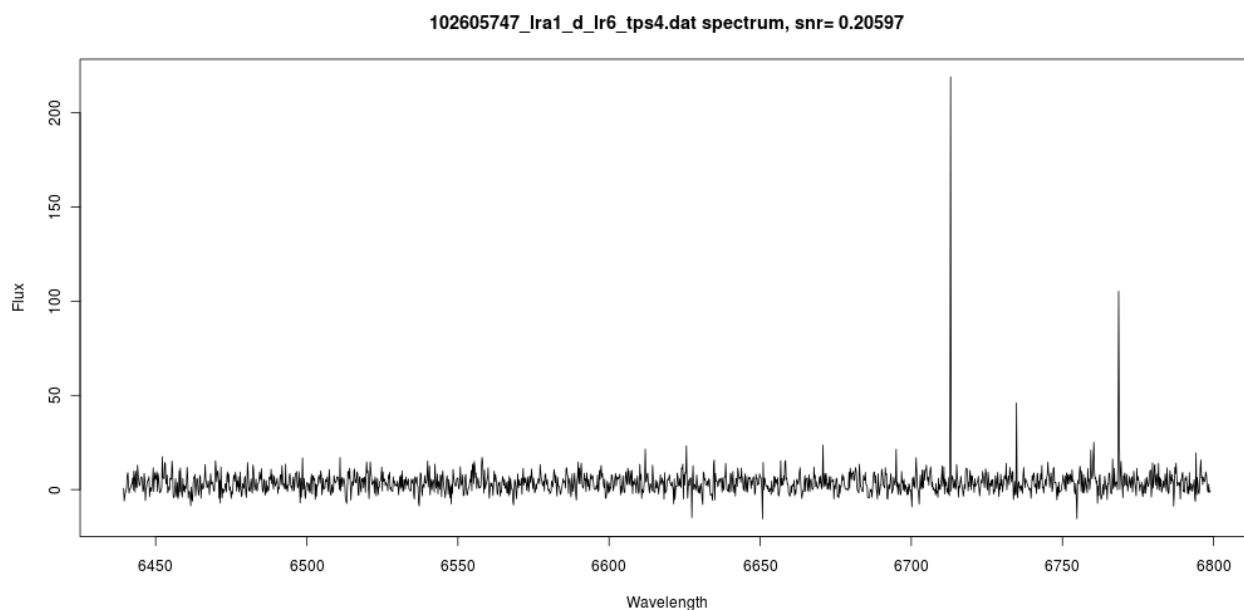


Ilustración 11: Espectro del FLAMES en el rango rojo con $SNR < 0.5$

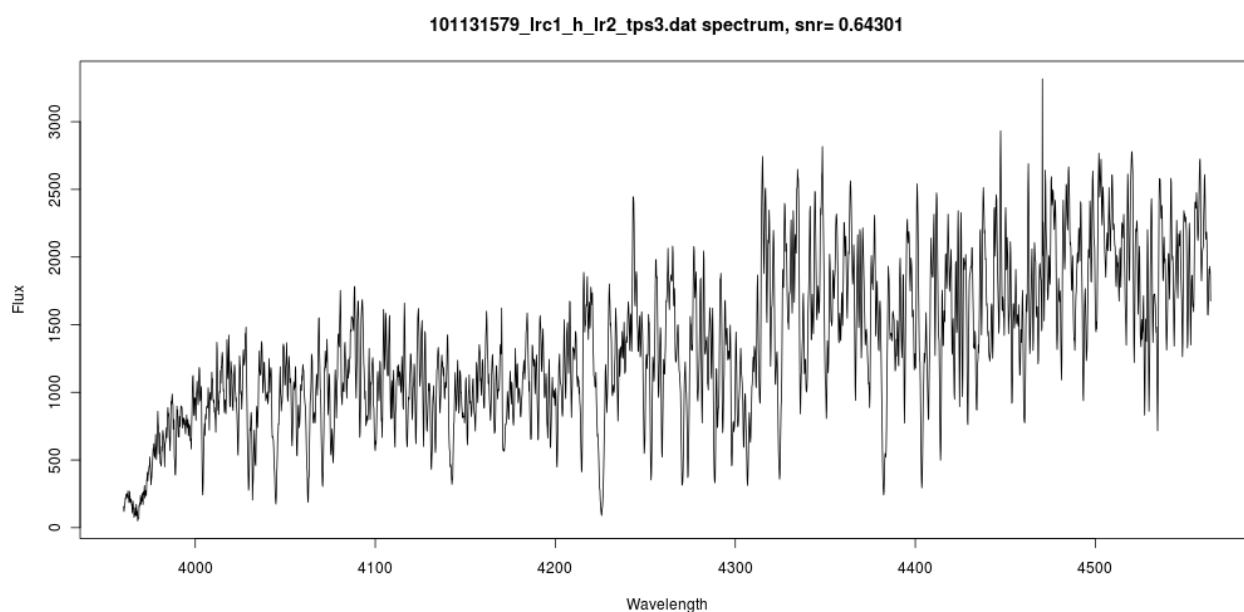


Ilustración 12: Espectro del FLAMES en el rango azul con $0.5 < SNR < 1$

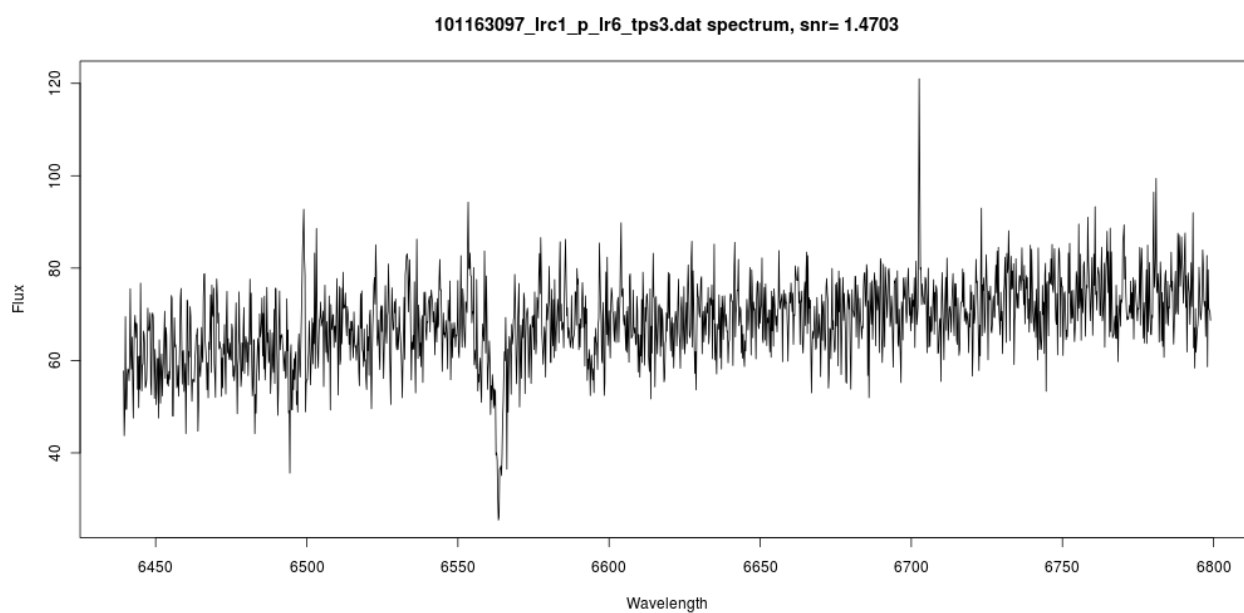


Ilustración 13: Espectro FLAMES en el rango rojo con $1 < SNR < 2$

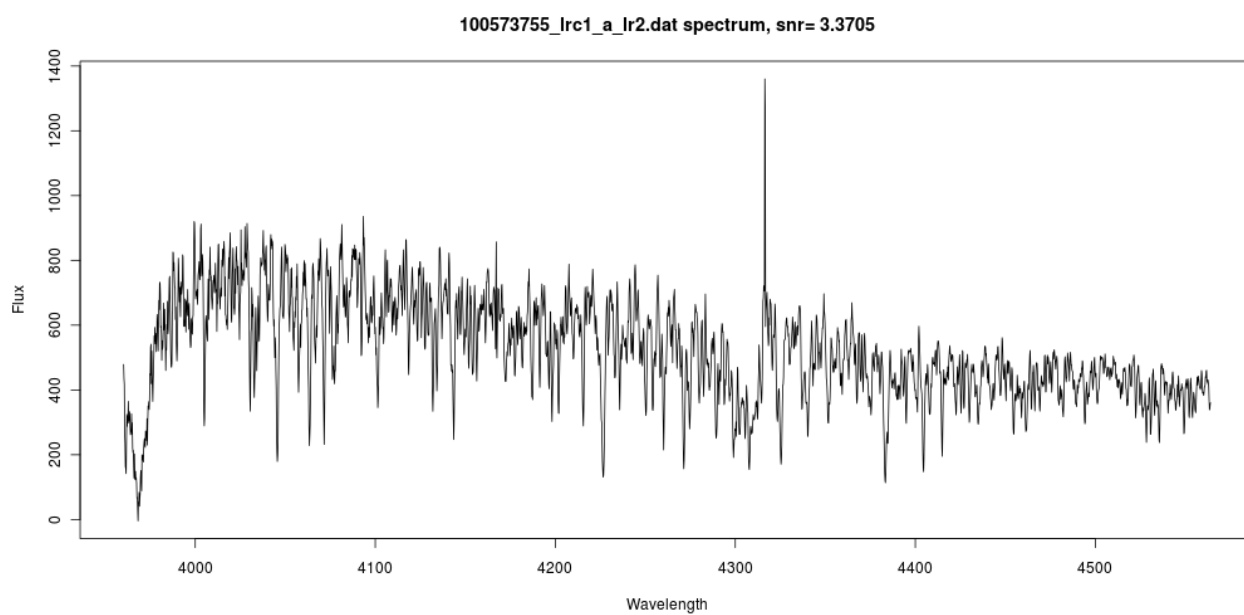


Ilustración 14: Espectro FLAMES en el rango azul con $3 < SNR < 4$

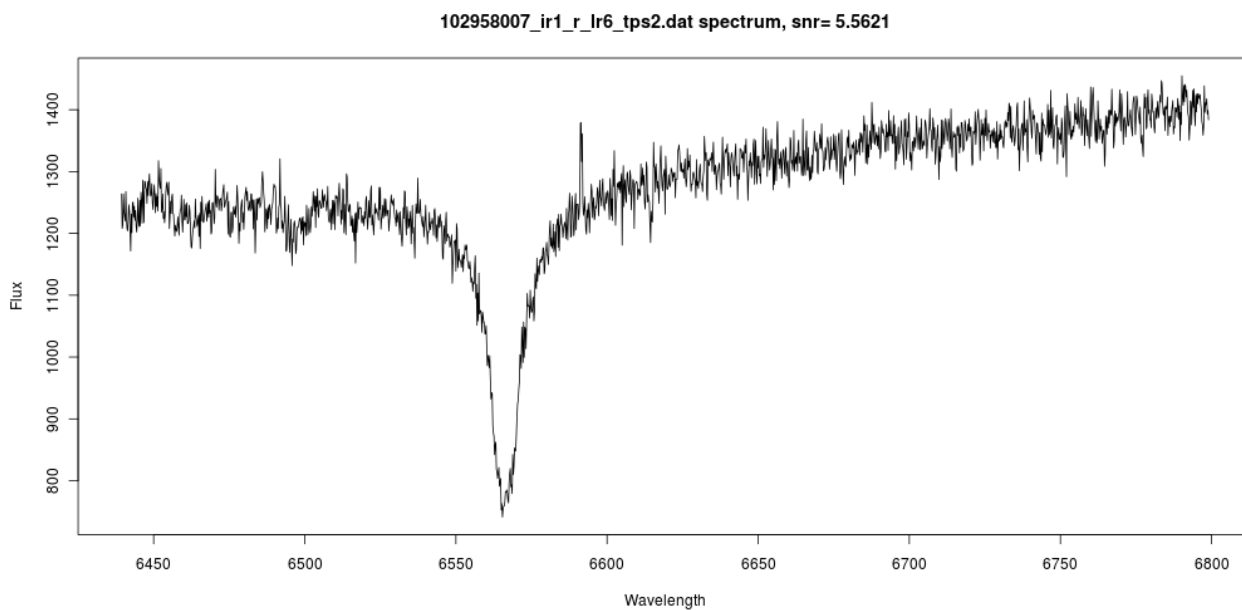


Ilustración 15: Espectro FLAMES en el rango rojo con $4 < SNR < 6$

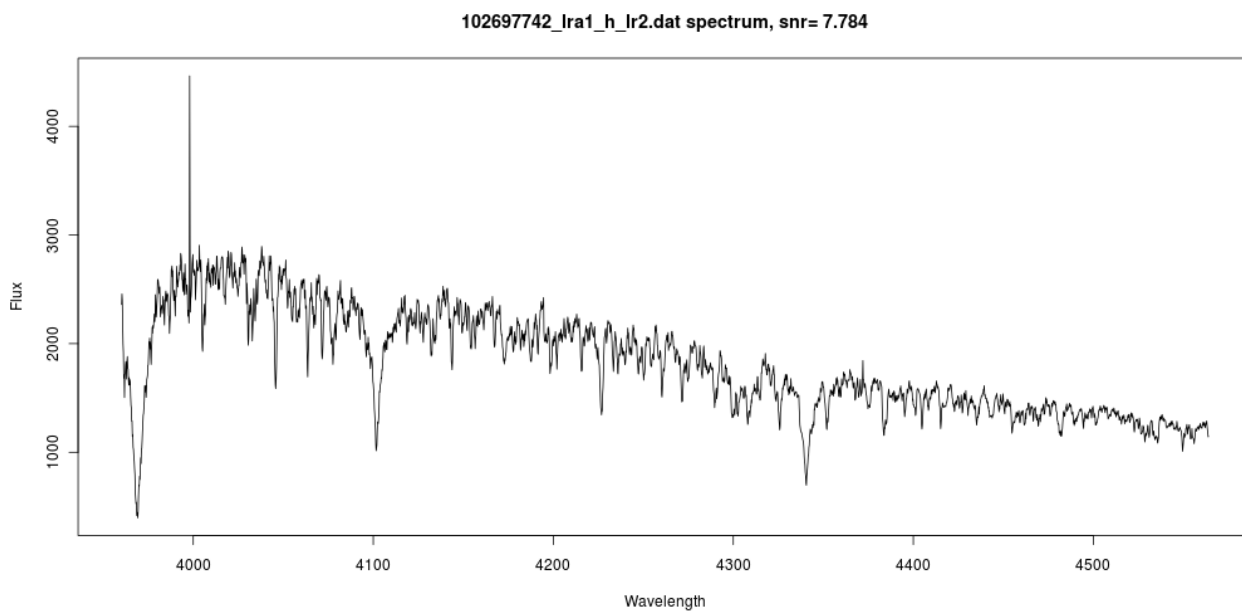


Ilustración 16: Espectro FLAMES en el rango azul con $6 < SNR < 8$

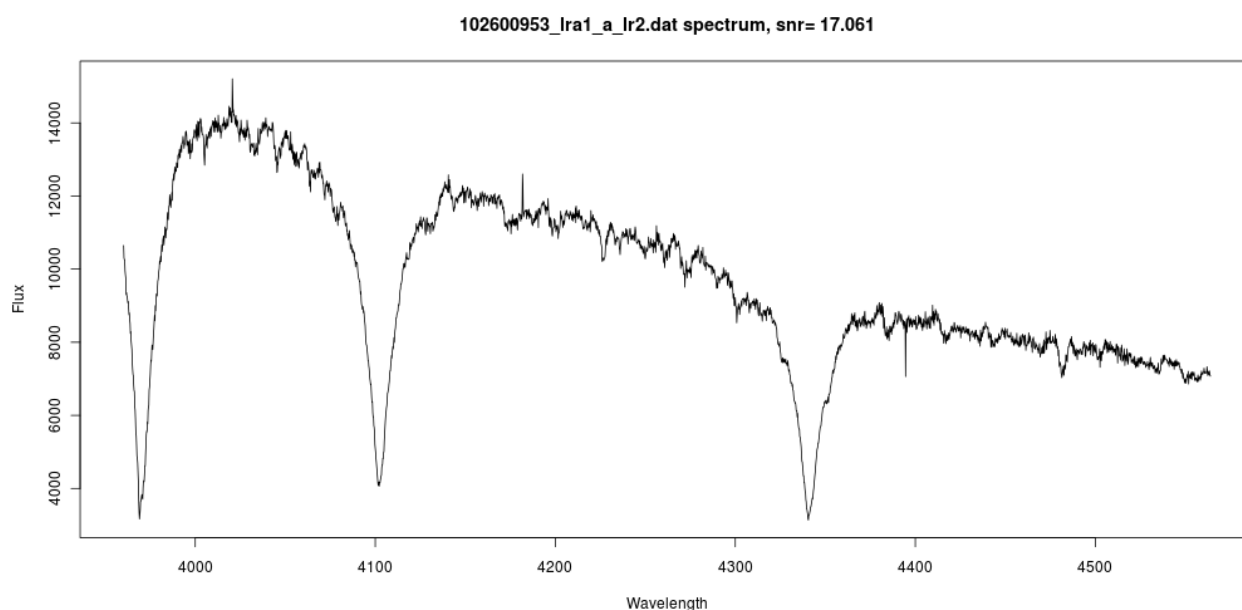


Ilustración 17: Espectro FLAMES en el rango azul con $SNR > 8$

Según se ha descrito, los espectros con bajo SNR deberían corresponder a estrellas con mucho ruido. Sin embargo, según la apreciación del tutor, podrían corresponder a estrellas de baja metalicidad. Por este motivo, en los experimentos realizados se utilizaron todos los espectros, independientemente del valor SNR obtenido.

3.2.6 Corrección del desplazamiento Doppler

Si las estrellas se desplazan a una velocidad con respecto a nosotros, cosa que pasa para la mayoría de ellas en menor o mayor medida, su espectro se desplaza con respecto al espectro que tendría si se mantuviesen siempre a la misma distancia. Este fenómeno se conoce como desplazamiento Doppler (*Doppler shift*(8)), y se manifiesta como un cambio de posición en las líneas de absorción respecto a la posición que tendrían esas líneas en el laboratorio. El desplazamiento viene dado por la fórmula:

$$\frac{v}{c} = \frac{\Delta\lambda}{\lambda}$$

donde v es la velocidad radial de la fuente, es decir, la velocidad de la fuente en la dirección en la que la se observa, c la velocidad de la luz, λ es la longitud de onda considerada y $\Delta\lambda$ es el desplazamiento respecto a λ . Cuando el objeto se acerca a la posición del observador, el espectro se desplaza a longitudes de onda menores, se dice entonces que está desplazado hacia el azul. Cuando el objeto se aleja respecto a la posición del observador, su espectro se desplaza hacia longitudes de onda mayores, y se desplaza hacia el rojo.

En las primeras pruebas con PCA, se observaron diferencias notables en los valores de las componentes principales entre el conjunto de datos de Kurucz y FLAMES I. Particularmente, se observó que, en muchos casos, las componentes principales de FLAMES no ocupaban el mismo espacio que sus correspondientes en los modelos de Kurucz.

Principal Components Diagram: log(Teff) distribution, red interval, Kurucz models and FLAMES stars

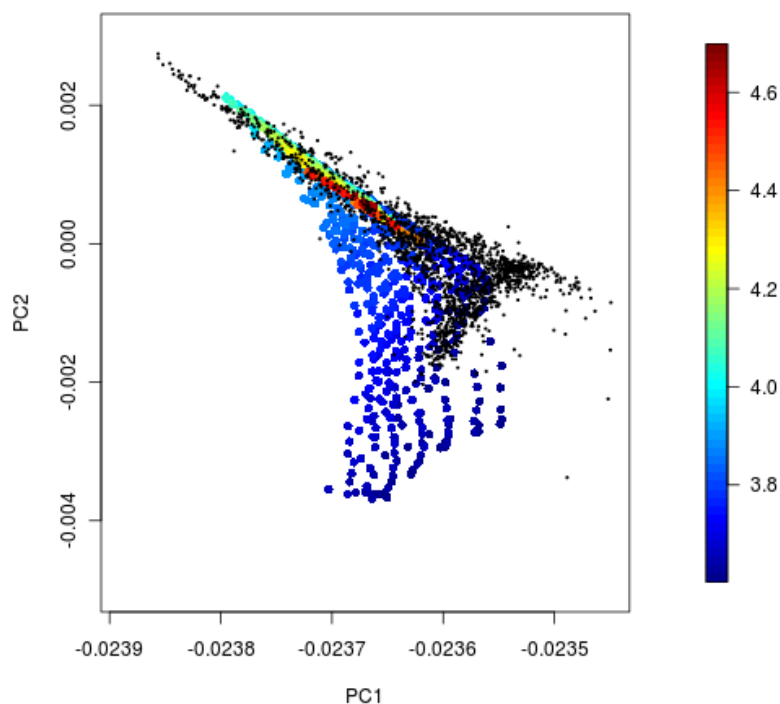


Ilustración 18: Distribución de estrellas de FLAMES I en PC1 y PC2.

Los modelos de Kurucz aparecen coloreados según el logaritmo de la temperatura, y las estrellas de FLAMES aparecen en color negro.

Una parte de las estrellas de FLAMES aparecen fuera del espacio que ocupa n los modelos de Kurucz, en la zona derecha de la gráfica.

En la Ilustración 18 se representan las dos primeras componentes principales de todos los espectros de Kurucz y de FLAMES. Los modelos de Kurucz aparecen coloreados según el logaritmo de la temperatura, y las estrellas de FLAMES aparecen en color negro. En la zona derecha de la gráfica se aprecia una zona donde se *amontonan* un buen número de espectros de FLAMES, en un espacio diferente del que ocupan los modelos de Kurucz.

Se estudiaron los espectros en el entorno de la longitud de onda 6563 Å, donde se encuentra una prominente línea de absorción en la mayoría de los espectros. Al compararlos con los modelos de Kurucz, se observó que una gran parte de ellos presentaba un desplazamiento en la línea de absorción. Este desplazamiento es debido al efecto Doppler ya comentado, lo que hizo suponer que este fenómeno estaba alterando los valores reales de las componentes principales. Una forma de corroborar este hecho fue calcular la desviación típica del índice cuyo valor es mínimo en ese entorno. Mientras que en los modelos de Kurucz, la desviación típica era 0.13, en un grupo de estrellas FLAMES dentro del espacio Kurucz era de 2.71, y para un grupo de estrellas FLAMES fuera del espacio Kurucz la desviación típica era de 10.71.

La prueba definitiva de este problema se muestra en la Ilustración 19, donde se representan nuevamente las dos primeras componentes principales de los modelos de Kurucz y las estrellas FLAMES, los primeros en color negro y las segundas coloreados. El significado del color es el desplazamiento de la línea de absorción en número de píxeles, siendo los azulados los desplazados a la izquierda, los amarillos y naranjas los que menos desplazamiento tienen y los rojos los desplazados a la derecha. Se puede observar cómo la mayoría de las estrellas FLAMES que caen fuera del espacio de los modelos de Kurucz tienen colores azules o rojos, mientras que la mayor cantidad de amarillos y naranjas se encuentra dentro del espacio de Kurucz.

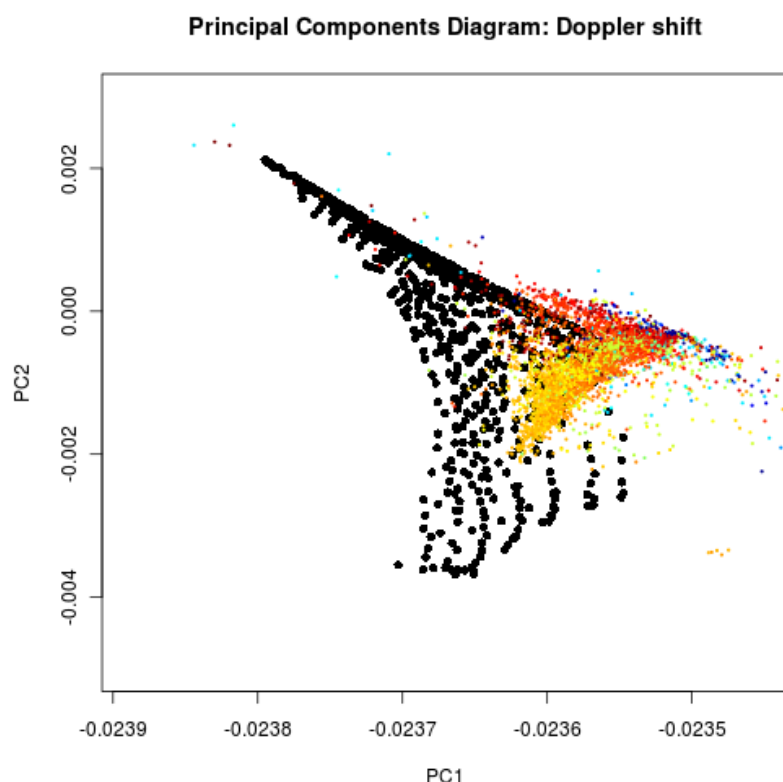


Ilustración 19: PC1 vs. PC2 de FLAMES I coloreados según su desplazamiento Doppler.

Los puntos azules y rojos corresponden a estrellas con gran desplazamiento por el efecto doppler. Los puntos amarillos y anaranjados a estrellas con poco o ningún desplazamiento doppler.

La forma de corregir este defecto fue la de comparar ambas líneas de absorción y adaptar el espectro FLAMES para que coincidieran las líneas de absorción. Para eso, había que buscar el método para identificar la línea de absorción y calcular su desplazamiento con respecto a una estrella del mismo tipo cuya velocidad radial fuera cero.

En el intervalo rojo, existe una línea de absorción en la longitud de onda de 6563 Å que se presenta en todos los modelos de Kurucz, en mayor o menor profundidad. En el intervalo azul, no existe una línea clara para cualquier temperatura. Sin embargo, entre las longitudes de onda 3964 Å y 3974 Å aparecen casi siempre sólo dos mínimos, en 3968,4 Å y en 3970 Å, el primero prevalece en modelos con baja temperatura y el segundo en los modelos con muy alta temperatura.

Respecto a la forma de identificar el desplazamiento, una forma sencilla que se encontró fue hallar el mínimo en las proximidades de la longitud de onda característica de una línea de absorción. Se entiende por “proximidades” unos 8 Å para el rango azul y 10.4 Å para el rojo a ambos lados de la longitud de onda característica de la línea de absorción considerada, que corresponden a 40 y 52 valores de flujo, respectivamente. Si un desplazamiento fuera mayor de 8 Å en el rango azul (10.4 en el rojo), supondría que la estrella tiene una velocidad radial superior a 600 Km/s (450 Km/s en el rojo) lo cual es bastante improbable, puesto que las estrellas hasta ahora observadas tienen velocidades que varían entre 0 y 400 Km/s, situándose la mayoría de ellas en el rango de entre los 10 y 40 Km/s.

El procedimiento fue el siguiente:

1. Se tienen en consideración dos grupos de valores de flujo cercanos a la longitud de onda *de referencia*, uno de los grupos es el doble que el otro y lo abarca.

2. Si en ambos grupos, el mínimo valor se encuentra en alguno de los extremos de ese grupo, no podemos tener la seguridad de dónde está la línea de absorción, el espectro no se modifica.
3. Si hay mínimos diferentes en los dos grupos, entonces se supone que existe dos líneas de absorción muy cercanas, como ocurre en el espectro azul. En este caso, se tiene en cuenta el mínimo del grupo más pequeño, por considerarse más probable que sea el más cercano a la línea de absorción de referencia, incluso si no es el menor de los dos mínimos. Se toma en cuenta el desplazamiento de ese mínimo respecto a la línea de absorción de referencia.
4. Si los mínimos coinciden en ambos grupos, entonces no hay duda. Se toma en cuenta el desplazamiento del único mínimo respecto a la línea de absorción de referencia.

Aplicando este sencillo algoritmo, solo se descartaron las estrellas cuyo mínimo se encontraba en alguno de los dos extremos del intervalo considerado, que en el intervalo rojo, fue solo de un 0,18%, y en el rango azul fue de 0.06%. Además, el 99,44% (en el rango azul) y el 99,39% (en el rango rojo) de los mínimos se encontraban desplazados a no más de 4 Å en el rango azul y 5,2 Å en el rojo.

Después de hallar el desplazamiento, se calculó por interpolación lineal cuál debía ser valor del flujo en cada longitud de onda de los modelos de Kurucz considerados, aplicando la fórmula que relaciona desplazamiento y longitud de onda expuesta al principio del presente apartado, y se volvieron a calcular las componentes principales de los espectros corregidos. La Ilustración 20 representa la posición de las dos primeras componentes principales de los espectros corregidos (puntos negros) y de los modelos de Kurucz (puntos coloreados). En dicha gráfica se puede observar visualmente que la mayoría de los espectros que caían fuera del espacio de los modelos de Kurucz ahora caen dentro.

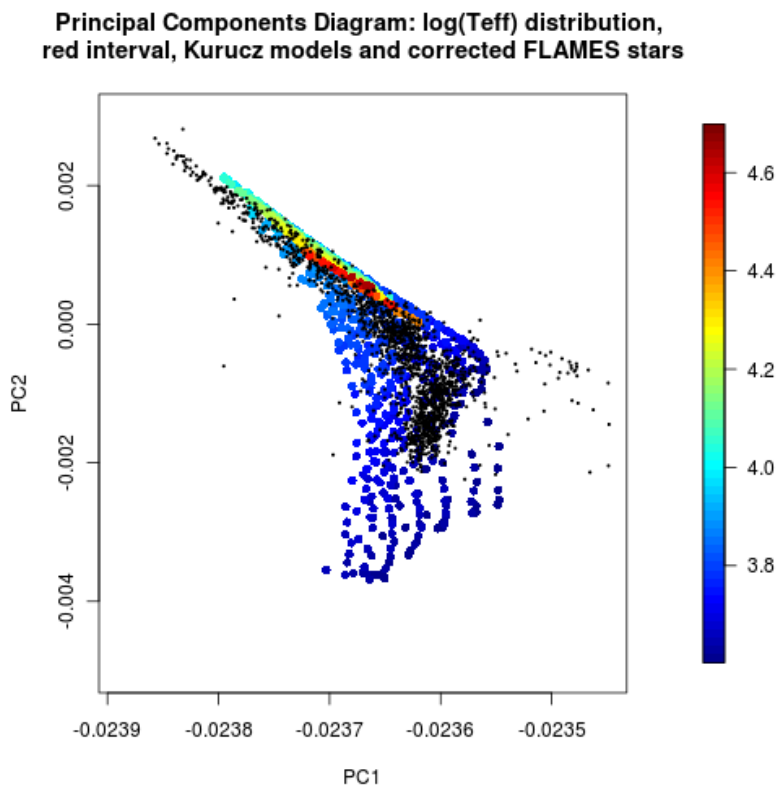


Ilustración 20: Estrellas de FLAMES I después de la corrección del desplazamiento doppler.

Los puntos coloreados corresponden a modelos de Kurucz, según su temperatura efectiva. Los puntos negros son de estrellas de FLAMES I a las que se les ha corregido el efecto doppler. Después de la corrección, la mayoría de las estrellas FLAMES se ajustan al espacio ocupado por los modelos de Kurucz.

3.2.7 Normalización

De las estrellas reales se reciben cantidades diferentes de energía en función de muchos factores, entre los que se encuentra la distancia o su masa. De dos estrellas similares, recibiremos más energía de la que más cerca se encuentre. Por este motivo, es necesario normalizar los datos para que puedan ser comparables.

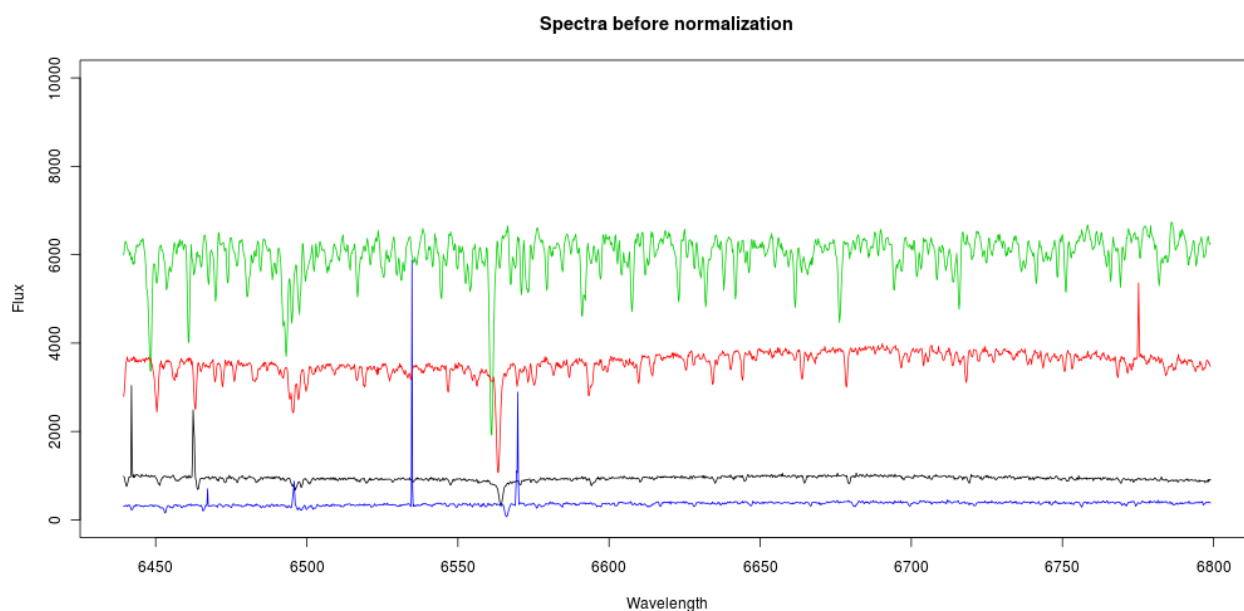


Ilustración 21: Varios espectros de FLAMES II antes de la normalización

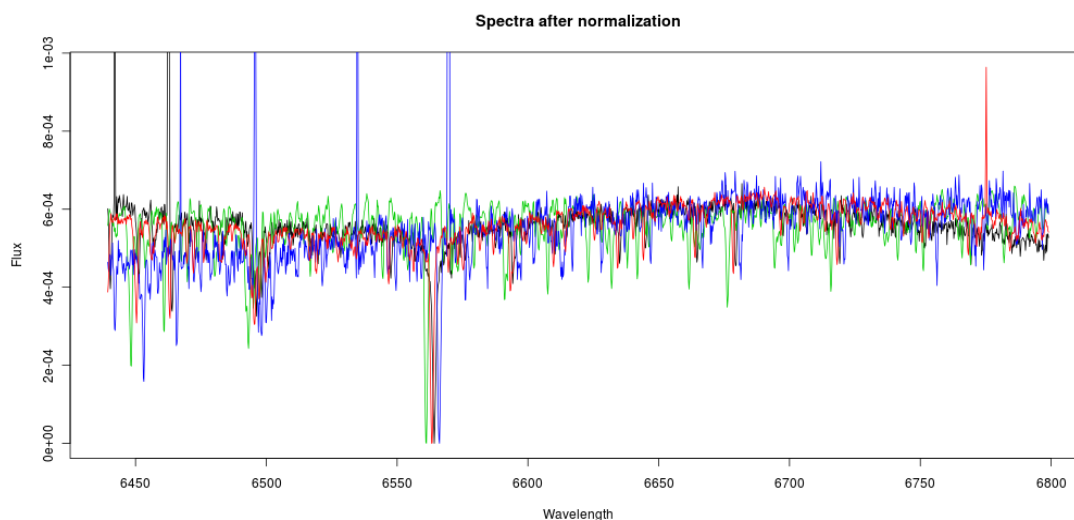


Ilustración 22: Espectros de FLAMES II después de la normalización

La forma que se eligió para normalizar los espectros fue la de igualar el área por debajo del espectro, de acuerdo a la siguiente fórmula:

$$f_N(w) = \frac{f(w) - \min(f(w))}{\sum f(w) - \min(f(w))}$$

donde,

$f_N(w)$ es el valor de flujo normalizado para la longitud de onda w ,

$f(w)$ es el valor de flujo medido para la longitud de onda w ,

$\min(f(w))$ es el valor mínimo de flujo para el espectro dado.

En la Ilustración 21 se muestran varios espectros del conjunto de datos de FLAMES II. La Ilustración 22 muestra los mismos espectros después de haber pasado por el filtro de la normalización.

3.3 Minería de datos

3.3.1 Análisis de componentes principales y máquinas de vectores soporte

Uno de los problemas que se encuentran al tratar de predecir parámetros físicos de la estrella a partir del espectro es la gran cantidad de datos que contiene un espectro. El enfoque de la solución propuesta es utilizar el análisis de componentes principales (en adelante, PCA) para reducir la dimensionalidad de los datos de entrada, y las máquinas de vectores soporte (en adelante, SVM) para obtener un modelo predictivo a partir de las componentes principales.

3.3.1.1 Análisis de componentes principales (PCA)

En apartados anteriores, se estableció que se iban a tener 1799 y 3020 valores de flujo por cada espectro, para el rango azul y rojo, respectivamente. Es evidente que tal número de valores de flujo se hace intratable para muchas técnicas de minería de datos y que se hace necesario un proceso previo de reducción de dimensionalidad.

PCA es una de las técnicas más conocidas de reducción de dimensionalidad. PCA obtiene, a partir de un conjunto de vectores, otro conjunto resultante del anterior transformado por un cambio de base. El conjunto de vectores resultante concentra la varianza en los primeros componentes de los vectores, y a estos se les denomina componentes principales (PC).

En R, la función que realiza el análisis de componentes principales sobre un conjunto de datos es *prcomp*, y se encuentra en el paquete *stats* de R. Aplicándola sobre los modelos de Kurucz, se observa que en los 30 primeros componentes principales, se acumula una varianza del 95,4% en el rango azul y del 98,9% en rango rojo. La Ilustración 23 muestra cómo crece la varianza acumulada según el número de componentes principales en ambos rangos.

Inicialmente, se pensó utilizar un número suficiente de PC para cubrir un porcentaje del 95% de la varianza. Finalmente, se decidió utilizar los 30 primeros PC en ambos rangos, ya que a partir de los experimentos realizados, no estaba claro que la varianza que acumulaban los primeros PC fuera un buen índice para escoger el número óptimo de PC. Este aspecto se expondrá con más detalle en el apartado 3.3.1.3.

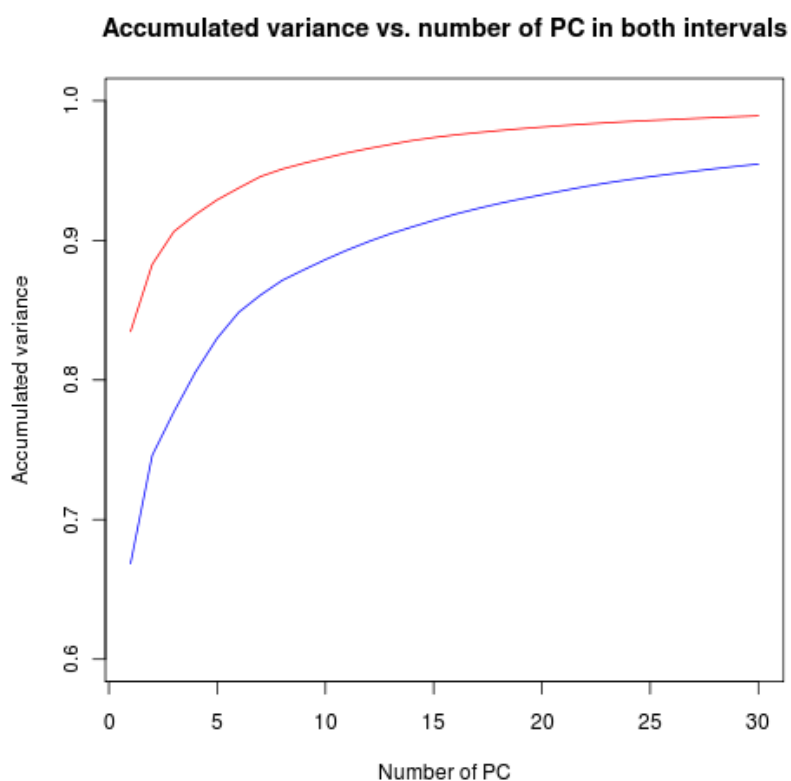


Ilustración 23: Varianza acumulada por número de componentes principales

3.3.1.2 Primeros experimentos de regresión con máquinas de vectores soporte (ϵ -SVM)

SVM es una técnica de clasificación lineal fundamentado en la búsqueda de separadores lineales en espacios vectoriales. SVM con margen máximo (13) es su versión más simple, y trata de encontrar un clasificador en un conjunto de datos de dos clases. A los datos de entrada se les considera vectores. La técnica busca un hiperplano $h(x)$ que separe linealmente el espacio de entrada en dos conjuntos. Un hiperplano puede expresarse de la siguiente forma:

$$h(x) = \langle w, x \rangle + b = 0$$

donde $h(x)$ es el hiperplano separador

w es el vector de pesos, un vector ortogonal al hiperplano

$\langle w, x \rangle$ es el producto escalar

b es el sesgo. $b/\|w\|$ determina el desplazamiento del hiperplano desde el origen.

SVM busca los valores de w y b que hacen que el hiperplano se encuentre a la misma distancia de los ejemplos más cercanos de cada clase. El clasificador se puede expresar como $\text{signo}(h(x))$, es decir, el signo de la función $h(x)$ decide si el vector x se encuentra a un lado o a otro del hiperplano, lo que determina su pertenencia a una clase u otra. Este problema se reduce a encontrar la solución al siguiente problema:

$$\begin{aligned} &\text{Minimizar } \frac{1}{2} \langle w, w \rangle \\ &\text{sujeto a } y_i (\langle w, x_i \rangle + b) \geq 1 \quad 1 \leq i \leq N \end{aligned}$$

Donde y_i es la variable de salida para el vector x_i de entrada, y N es el número total de vectores de

entrada.

El problema, tal y como se ha planteado, puede no tener solución, pues exige que los datos de entrada sean linealmente separables. Este inconveniente hace que normalmente se utilice una variación del algoritmo: SVM con margen blando. La idea principal es la introducción de variables de holgura que permiten que la restricción de separación lineal no se cumpla de forma estricta. El problema queda transformado de la siguiente forma

$$\begin{array}{ll} \text{Minimizar} & \frac{1}{2} \langle w, w \rangle + C \sum_{i=1}^N \xi_i \\ \text{sujeto a} & y_i (\langle w, x_i \rangle + b) \geq 1 - \xi_i \quad 1 \leq i \leq N \\ & \xi_i \geq 0 \quad 1 \leq i \leq N \end{array}$$

Las variables de holgura se denotan con la letra griega ξ , y deben tener valores entre 0 y 1. C es una constante que permite ajustar el margen blando; a mayores valores de C , menor es la tolerancia a permitir vectores de entrada que no se encuentren en dentro del espacio que corresponde a su clase.

En realidad, SVM con margen blando no es aplicable al problema propuesto debido a que ni es un problema de clasificación, sino de regresión, y tampoco está claro que el problema sea linealmente separable. Sin embargo, existen variantes de esta técnica que permiten solventar los inconvenientes anteriores:

- El problema de la separabilidad lineal se resuelve mediante el uso de transformaciones no lineales del espacio de entrada a un espacio dotado de un producto escalar (el espacio de características) en el que los datos sí sean linealmente separables. Las funciones que hacen este tipo de transformaciones se conocen como funciones núcleo, o *kernels*.
- Hay que utilizar una variación de SVM para regresión, conocida como ϵ -SVM, y cuyo algoritmo busca una función no lineal de manera que todos los datos de aprendizaje estén a una distancia menor o igual que ϵ . Ambas variantes se explicarán a continuación.

La variante de SVM utilizando funciones núcleo (*kernel*) permite utilizar SVM sobre una transformación del espacio de entrada. Al espacio transformado lo denominaremos espacio de características. Se trata de buscar una separación lineal sobre el espacio de características, lo cual puede suponer una separación no lineal en el espacio de entrada. La búsqueda del hiperplano antes referido solo depende de la existencia de un producto escalar en el espacio de entrada, es decir, no es necesario trabajar directamente con los vectores transformados en el espacio de características, sino con los productos escalares entre dichos vectores. Si se encuentra una transformación no lineal del espacio de entrada a un espacio dotado de producto escalar (el espacio de características), se pueden aplicar los mismos razonamientos para hallar el hiperplano separador en el espacio de características.

La función que transforma dos vectores del espacio de entrada en un producto escalar de vectores del espacio de características es la función núcleo o *kernel*. Sea $\Phi: \mathbb{R}^D \rightarrow \mathfrak{F}$ la transformación del espacio de entrada en el espacio de características. Entonces, la función núcleo $K(x,y)$ es aquella que puede expresar el producto escalar de los vectores en el espacio de características como una función dependiente de los vectores en el espacio de entrada:

$$K(x, y) = \langle \phi(x), \phi(y) \rangle$$

Donde finalmente reside la utilidad de esta transformación es en encontrar un espacio de características donde los vectores transformados sí sean linealmente separables.

En el paquete *kernelab* de R, se permite utilizar SVM con diferentes kernels predefinidos o incluso

con kernels definidos por el usuario de la librería. Se realizaron pruebas con los kernels siguientes:

RBF (<i>Gaussian Radial Basis Function</i>)	$K(x, y) = \exp(-\sigma \ x - y\ ^2)$
Laplace Radial Basis Kernel	$K(x, y) = \exp(-\sigma \ x - y\)$
Bessel	$K(x, y) = \frac{Bessel^{n(v+1)}(\sigma \ x - y\)}{(\ x - y\)^{-n(v+1)}}$
PUK (<i>Pearson VII Universal Kernel</i>)	$K(x, y) = \frac{1}{\left(1 + \left(\frac{2 \cdot \ x - y\ \cdot \sqrt{2^{1/\omega} - 1}}{\sigma}\right)^2\right)^\sigma}$

La otra variante del algoritmo referida anteriormente es ϵ -SVM (14), que permite utilizar SVM como técnica de regresión. Es compatible con el uso de *kernels*. La regresión se realiza a partir de una función $f(x)$ que tenga, como mucho, una desviación ϵ con respecto al valor real. Esta función es una combinación lineal de los valores de entrada:

$$f(x) = \langle w, x \rangle + b$$

Como en el caso de la clasificación, puede ocurrir que dicha función no exista, porque no pueda cumplirse para la totalidad de los vectores de entrada. De nuevo, esta restricción puede suavizarse mediante SVM con margen blando, que introducía las variables de holgura (ξ, ξ^*). La formulación del problema con variables de holgura quedaría de la siguiente forma:

$$\begin{aligned} \text{Minimizar} \quad & \frac{1}{2} \langle w, w \rangle + C \sum_{i=1}^N (\xi_i + \xi_i^*) \\ \text{sujeto a} \quad & y_i - f(x_i) \leq \epsilon + \xi_i \quad 1 \leq i \leq N \\ & f(x_i) - y_i \leq \epsilon + \xi_i^* \quad 1 \leq i \leq N \\ & \xi_i, \xi_i^* \geq 0 \quad 1 \leq i \leq N \end{aligned}$$

El significado de ϵ es el siguiente: Las desviaciones de la función $f(x_i)$ con respecto a su valor real y_i se consideran aceptables si no superan ϵ , y solo se intenta minimizar el término $1/2 \langle w, w \rangle$. Las desviaciones que superan el margen de ϵ deben minimizarse junto a el término $1/2 \langle w, w \rangle$. El aumento de ϵ supone aumentar la tolerancia del error. La constante C tiene el mismo significado que en la fórmula de la clasificación.

Ya se ha visto que ϵ -SVM y la utilización de kernels permiten aplicar esta técnica al problema planteado en este trabajo fin de máster. Las dos variantes del algoritmo introducen parámetros que deben ser ajustados para encontrar el modelo regresivo óptimo. Las funciones núcleo suelen incluir uno o varios parámetros en su definición, tal es el caso de la constante en σ en el kernel RBF, o las constantes σ y ω en el kernel PUK. Por otra parte, el valor de ϵ en el algoritmo ϵ -SVM es otro parámetro que hay que ajustar. El valor de ϵ determina el nivel de precisión de la función regresiva, aunque esa precisión solo es asegurada en el conjunto de entrenamiento. Cuanto mayor es el valor de ϵ , menor número de vectores soporte se utilizan en la función. Valores altos de ϵ no pueden dar buenos resultados por utilizar pocos vectores soporte, y valores cercanos a cero pueden producir un sobreajuste de la función de regresión. Por tanto, el valor óptimo de ϵ depende en gran medida de los conjuntos de entrenamiento y de validación elegidos.

Para ajustar los parámetros óptimos de ϵ -SVM se construyeron las funciones en R que utilizaban distintos valores de ϵ y distintos valores de los parámetros de los diferentes kernel. Se desarrolló un procedimiento que calculaba los errores cometidos en una validación cruzada de 10 grupos para cada combinación de los parámetros del kernel. Los primeros resultados se muestran en la Tabla 2. Los errores registrados son el índice de correlación (Corr), el error medio absoluto (MAE), la raíz del error cuadrático medio (RMSE), en el apéndice I se encuentran sus definiciones. Para el kernel PUK se utilizó $\sigma=\omega=1$, ya que eran los valores que mejor resultados daban con la herramienta Weka. Para el resto, se buscó el mejor valor de σ , que aparece como una fila más en la tabla. Se observa que son los kernels RBF y de Laplace los que mejores índices de errores presentan, y no existen muchas diferencias entre ellos. Bessel y PUK pueden ser similares en algunos casos a los kernels de Laplace y RBF, pero en otros son claramente peores, así que fueron descartados.

		RANGO AZUL		RANGO ROJO	
		Teff	log(g)	Teff	log(g)
RBF	σ	0,07812	0,01563	0,5	0,25
	Corr	0,9926	0,9862	0,9837	0,9478
	MAE	343,42	0,1284	631,59	0,2762
	RMSE	1114,67	0,2481	1685	0,4976
Laplace	σ	0,125	0,0625	0,5	0,5
	Corr	0,9937	0,9898	0,9833	0,9637
	MAE	428,21	0,1292	583,78	0,2302
	RMSE	1208,93	0,2354	1730,3	0,4194
Bessel	σ	0,25	0,25	0,5	0,5
	Corr	0,9933	0,9788	0,8842	0,9013
	MAE	340,26	0,1641	1324	0,4377
	RMSE	1063,39	0,3073	4106	0,6607
PUK	σ	-	-	-	-
	Corr	0,64	0,75	0,9799	0,9285
	MAE	3173	0,8896	852,51	0,3606
	RMSE	6577	1,184	1989	0,6143

Tabla 2: Errores encontrados por kernel y rango

Si se analizan con más detalle los modelos generados, se pueden observar algunas peculiaridades. La comparación gráfica de las temperaturas predichas frente a las reales se muestra en la Ilustración 24. Nótese que en esta gráfica aparecen los valores de estrellas del rango rojo en rojo, y las del rango azul en azul. Se observa claramente que la predicción de la temperatura es peor a partir de los 10.000 K, y mucho peor a partir de los 20.000 K. La comparación numérica de los índices de error entre el conjunto de entrenamiento completo y sólo el conjunto cuya predicción es menor o igual a 20.000K es igualmente reveladora, véase la Tabla 3. Nótese la gran reducción que se produce en la raíz del error cuadrático medio.

	Teff, RANGO AZUL		Teff, RANGO ROJO	
	Completo	$\leq 20000\text{K}$	Completo	$\leq 20000\text{K}$
Corr	0,9944	0,9994	0,9839	0,9934
MAE	216,76	48,82	409,3	109,3
RMSE	972,69	139,32	1570,5	466,36

Tabla 3: Errores del conjunto completo y el conjunto con temperaturas ≤ 20.000 . Se utilizó el kernel RBF, para los modelos de Kurucz.

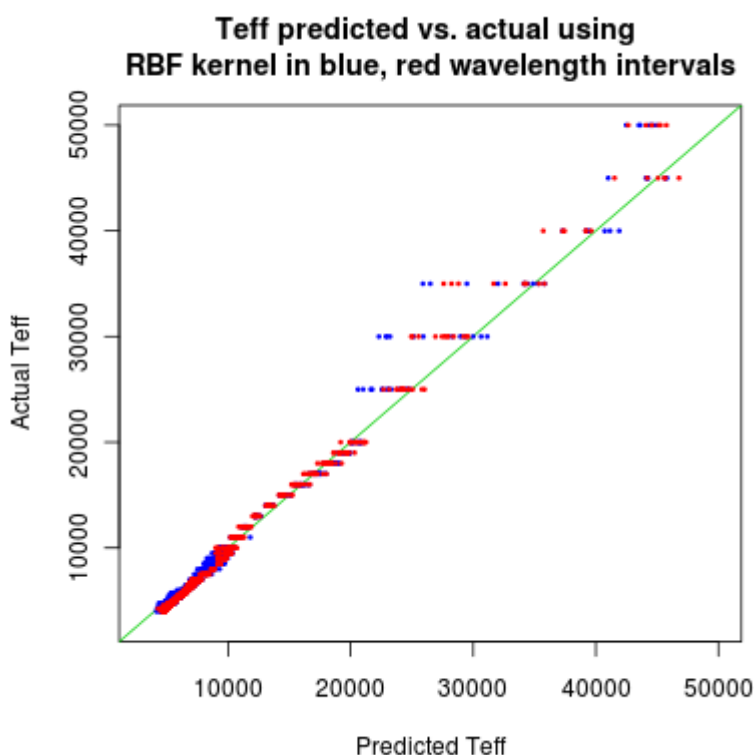


Ilustración 24: Temperatura efectiva vs. real utilizando PCA+SVM con el kernel RBF.

Se observa claramente que la predicción de la temperatura es peor a partir de los 10.000 K, y mucho peor a partir de los 20.000 K.

Como consecuencia de la diferencia de resultados entre el conjunto completo y el conjunto de estrellas menor de una determinada temperatura, se decidió que, en aras de buscar un modelo suficientemente preciso, había que restringir el número de estrellas por su temperatura. Se decidió que los modelos de Kurucz quizá no fueran suficientemente buenos para predecir los parámetros físicos de las estrellas a partir de los 10.000K. El hecho de que el número de modelos de Kurucz a partir de los 10.000 K fuera más escaso (y a partir de los 20000 K mucho más escaso) fue otra de las razones para tratar de no utilizarlos a partir de esas temperaturas⁶.

Sin embargo, la decisión de descartar los modelos de Kurucz superiores a 10.000 K trajo consigo un segundo problema, y es que había que tener algún modo de determinar que las estrellas reales que se iban a introducir en el modelo también tuvieran temperaturas inferiores a 10.000 K. Por ese motivo, era necesario crear un clasificador dicotómico que determinase si una estrella real estaba por debajo o por encima de los 10.000 K.

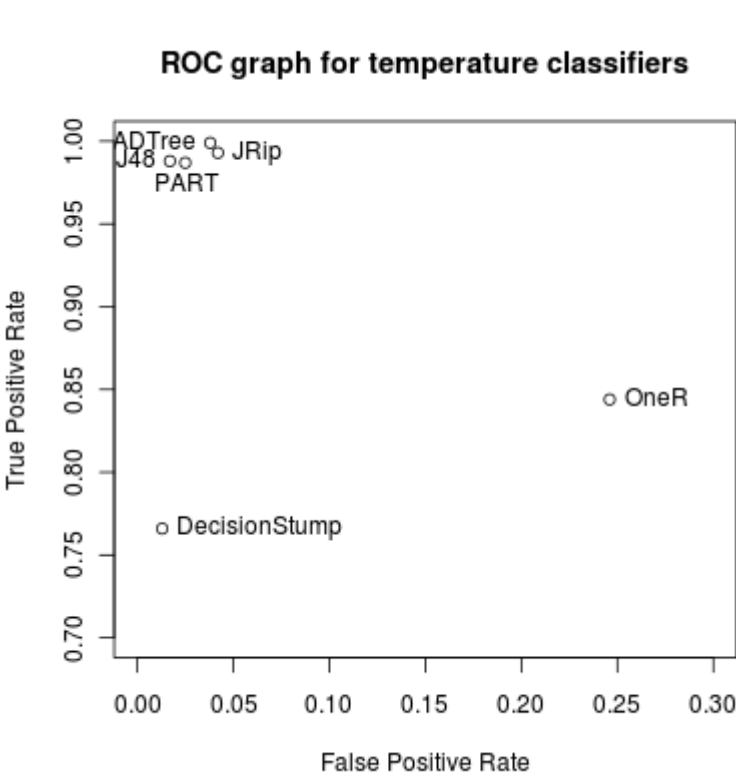
⁶ Para tratar de predecir los parámetros físicos de estrellas a partir de los 10.000 K, se realizaron pruebas con los modelos TLUSTY, unos modelos espectrales sintéticos para estrellas calientes; desgraciadamente los resultados no fueron buenos, véase el apartado 4.6.

3.3.1.3 Clasificador dicotómico

En trabajos anteriores se descubrió que los modelos desarrollados eran más imprecisos a partir de los 10.000K, por lo que se decidió desarrollar modelos que solo trabajaran con estrellas por debajo de esa temperatura. Para tal propósito, se debía desarrollar un clasificador dicotómico que determinase si una estrella dada se encontraba por debajo o por encima de los 10.000K con una precisión suficiente. Se considerará como clase positiva la que se compone de las estrellas cuya temperatura efectiva esté por debajo de 10.000 K, y la negativa la de las estrellas de temperatura superior a 10.000K.

La búsqueda de este clasificador se realizó utilizando dos herramientas de minería de datos, R y WEKA. Inicialmente, se buscó un modelo utilizando las 30 componentes principales en R a través de PCA, pero los resultados no fueron demasiado buenos para los datos en ELODIE. Weka es una herramienta que permite utilizar multitud de clasificadores sin mucho esfuerzo, así que se probó a crear clasificadores de técnicas muy diferentes.

	FPR	FNR
DecisionStump	0,013	0,234
J48	0,017	0,012
JRip	0,042	0,007
OneR	0,246	0,156
PART	0,025	0,013
ADTree	0,038	0,001



ROC graph for temperature classifiers

The graph plots True Positive Rate (Y-axis, 0.70 to 1.00) against False Positive Rate (X-axis, 0.00 to 0.30). Data points are labeled: ADTree (top-left), J48 (top-left), JRip (top-left), PART (top-left), OneR (middle-right), and DecisionStump (bottom-left).

Tabla 4: FPR y FNR en el rango azul para diferentes algoritmo de clasificación binaria. A la derecha, se muestra la gráfica ROC de los clasificadores estudiados. Por motivos de claridad, se muestra solo la esquina superior izquierda de la gráfica ROC

En la Tabla 4 se muestran el ratio de falsos positivos (FNR) y el ratio de falsos negativos (FNR) utilizado diversos algoritmos de clasificación en el rango azul de longitudes de onda, utilizando validación cruzada de 10 subconjuntos. En esta misma tabla, se muestra la gráfica ROC de los clasificadores estudiados. La gráfica ROC representa el ratio de falsos positivo (FPR) frente al ratio de positivos verdaderos (TPR). Permite distinguir de un golpe de vista la bondad de los clasificadores. La escala de la gráfica es de 0 a 1 tanto en las abscisas como en la ordenadas, pero,

por motivos de claridad, se ha optado por representar solo la zona próxima a la esquina superior izquierda de la gráfica. En las gráficas ROC, un clasificador es tanto más bueno cuanto más cerca se encuentre de la esquina superior izquierda. Los clasificadores que se acercan al lado izquierdo, y no tanto al superior, son buenos clasificadores para de la clase positiva y no tanto de la negativa. De forma análoga, los clasificadores que se acercan al lado superior y no tanto al lado izquierdo, son buenos clasificadores de la clase negativa y no tanto de la positiva. Antes de analizar los resultados, conviene que se aclaren algunos aspectos.

- 1) Los clasificadores anteriores tenían como conjunto de entrada las 30 primeras componentes principales del rango azul. Los experimentos con clasificadores en el rango rojo ofrecieron unos resultados muy pobres. Por otra parte, el conjunto de datos de validación es ELODIE, que es un conjunto de espectros que abarca ambos rangos de longitud de onda. Para realizar las pruebas hubo que dividir ELODIE en dos conjuntos, uno por cada rango de longitud de onda considerada. Estos dos conjuntos resultantes contienen, por lo tanto, las mismas estrellas, y en el mismo orden. Consiguientemente, no fue necesario un clasificador para cada rango de longitudes de onda. Bastaba un buen clasificador en alguno de los rangos de longitud de onda (el azul, en este caso) para clasificar tanto las estrellas del rango azul, como las del rango rojo.
- 2) El objetivo del clasificador que se estaba buscando era seleccionar estrellas por debajo de 10000K (estrellas frías). Es preferible que se descarten estrellas frías, a que se seleccionen estrellas calientes, debido a que el grupo de estrellas calientes no va a ser utilizado en posteriores estudios. Esto significa que lo que se persigue es un clasificador con muy pocos falsos positivos, aunque el número de falsos negativos no sea tan bajo. Volviendo a la gráfica ROC, el clasificador buscado es aquel que se acerca más al lado izquierdo de la gráfica, y que esté también cerca del lado superior, aunque haya otros que estén más cerca del lado superior. Es preferible descartar unas pocas estrellas frías, que incluir en el conjunto de estrellas frías un número mayor de estrellas calientes.

Hechas estas consideraciones, si se observa la gráfica ROC anterior, se aprecia que el mejor clasificador es *J48*, aunque es *DecisionStump* el que presenta mejor FPR. En este punto del estudio, conviene profundizar en los algoritmos de los dos clasificadores: *DecisionStump* y *J48*.

DecisionStump (15) es un algoritmo que construye un sencillo árbol de un solo nivel con uno solo de los atributos de entrada. El algoritmo busca el atributo que tiene mayor probabilidad de predecir la clase, y encuentra una regla de asociación utilizando dicho atributo para predecir la clase. En los experimentos, la regla fue:

Si $(PC2 \leq 0.001302)$ entonces clase ← estrella fría
 en otro caso clase ← estrella caliente

Lo interesante de esta regla es que, a pesar de utilizar un solo atributo, esta sencilla regla tiene el más bajo FPR de todos. Además, cabe destacar que la primera componente principal no ha sido seleccionada para hacer la regla, a pesar de ser el atributo de entrada que mayor varianza acumula.

El segundo algoritmo de clasificación es *J48*, implementación en Weka del algoritmo *C4.5*. Este algoritmo utiliza un árbol de decisión para estimar la clase (caliente, fría) de la estrella. Un árbol de decisión es un conjunto de condiciones organizadas en una estructura jerárquica, de tal manera que la estimación final se encuentra siguiendo las condiciones que se cumplen desde la raíz del árbol hasta alguna de sus hojas. En Weka, el árbol hallado por *J48* es el siguiente:

```

PC2 <= 0.001266: cold (454.0)
PC2 > 0.001266
| PC8 <= -0.000081
| | PC13 <= -0.000069
| | | PC1 <= -0.018386: cold (2.0)
| | | PC1 > -0.018386: hot (4.0/1.0)
| | PC13 > -0.000069
| | | PC20 <= -0.000096
| | | | PC1 <= -0.018396: cold (4.0)
| | | | PC1 > -0.018396: hot (3.0)
| | | PC20 > -0.000096: cold (85.0)
| PC8 > -0.000081
| | PC6 <= -0.000082
| | | PC11 <= -0.000265
| | | | PC8 <= 0.000064: cold (11.0)
| | | | PC8 > 0.000064: hot (2.0)
| | | PC11 > -0.000265
| | | | PC7 <= 0.000065
| | | | | PC18 <= -0.000062
| | | | | | PC8 <= 0.000012: cold (11.0)
| | | | | | PC8 > 0.000012: hot (7.0/1.0)
| | | | | PC18 > -0.000062: hot (45.0)
| | | | | PC7 > 0.000065: hot (165.0)
| | PC6 > -0.000082
| | | PC8 <= 0.000039: cold (26.0)
| | | PC8 > 0.000039: hot (12.0)

```

El árbol tiene 6 niveles, aunque las hojas no están todas en el último. Por ejemplo, una estrella se considera fría sin $PC2 \leq 0.001266$, y se considera caliente si $PC2 > 0.001266$, $PC8 \leq -0.000081$, $PC13 \leq -0.000069$, y $PC1 > -0.018386$.

Sin embargo, hay algo que merece ser tenido en cuenta en el árbol de decisión anterior: solo utiliza 9 de las 30 componentes principales para conseguir el mejor árbol. Además, la importancia de cada atributo es mayor cuanto más cerca se encuentra el nodo de decisión de la raíz del árbol; nótese que PC7 solo es requerida cuando PC2, PC8, PC6 y PC11 se encuentran en un determinado rango, mientras que PC2 participa en todas las decisiones. Se observa que PC1, a pesar de ser la componente principal que mayor varianza acumula, tiene una importancia más bien baja ya que aparece en el tercer y cuarto nivel del árbol y solo participa en 4 de las 14 decisiones del árbol.

A partir de la búsqueda de clasificadores con la herramienta Weka, se ha encontrado una selección de atributos que podría dar mejores resultados en otros clasificadores. Los atributos del árbol de decisión más relevantes desde la raíz a las hojas son PC2, PC8, PC6, PC13, PC11, PC20, PC1, PC7, y PC18 por orden de relevancia. Con esta selección de atributos, se volvió a utilizar R para buscar un clasificador utilizando PCA, esta vez con los atributos seleccionados de WEKA. La razón de utilizar R fue la de estudiar el valor de los falsos positivos y falsos negativos, algo que no se puede hacer con Weka. Como se conocía que el PC2 era un buen atributo clasificador por sí solo, se decidió crear un conjunto de clasificadores con los PC anteriores, pero incorporándolos de forma progresiva, y por orden de relevancia, es decir, primero el conjunto (PC2), luego (PC2, PC8), luego (PC2, PC8, PC6) y así sucesivamente. Para hacer más fiable la predicción, esta vez se realizó la validación con el conjunto de datos de ELODIE. La Tabla 5 ofrece el FPR y FNR, y las medias de las temperaturas de los falsos positivos (FP) y falsos negativos (FN).

Si se analiza la tabla, se comprueba que, según el FPR y el FNR, el mejor clasificador solo contiene la PC2 en solitario. Sin embargo, aun siendo un buen clasificador, los falsos positivos poseen una temperatura muy superior a la que sería deseable. En cambio, tomando como medida de bondad las

medias de los falsos positivos y los falsos negativos, el mejor clasificador es el de cuatro componentes principales. Después de estudiar las temperaturas de los falsos positivos⁷, se encontró que, a partir de los clasificadores de dos componentes, abundaban estrellas que tenía casi 10000K, concretamente, 10012K. Esta leve diferencia no puede considerarse un error, así que se volvió a calcular de nuevo los FPR y FNR, utilizando el límite de temperatura de los 10013K tanto en el conjunto de entrenamiento como en el de validación. Los resultados se muestran en la Tabla 6.

PC	FPR	FNR	Media Teff FP	Media Teff FN
2	0,08391	0,06232	35228	7878
2,8	0,2167	0,0155	18366	7999
2,8,6	0,2167	0,0128	18366	8096
2,8,6,13	0,3426	0,00824	16306	8693
2,8,6,13,11	0,3496	0,009	18756	8590
2,8,6,13,11,20	0,3216	0,0238	18689	7834
2,8,6,13,11,20,1	0,3216	0,024	19201	7664
2,8,6,13,11,20,1,7	0,3006	0,018	20009	7816
2,8,6,13,11,20,1,7,18	0,4055	0,004	18465	8663

Tabla 5: Clasificación de estrellas frías (<10000 K) y calientes (>=10000K). Se muestra FPR, FNR, media de Teff en los falsos positivos y en los falsos negativos, en el rango azul, utilizando PCA y SVM como clasificador.

Se aprecia que el clasificador con las componentes principales PC2, PC8 y PC6 obtiene los mejores FPR y FNR, que son mejores que los mejores encontrados en la Tabla 5. La lista de temperaturas de los falsos positivos ya no contiene valores muy cercanos a 10012. En conclusión, utilizando ELODIE como conjunto de validación, el clasificador dicotómico que utiliza las PC (2,8,6) y SVM es el mejor clasificador, y es el que se utilizará para seleccionar las estrellas de ELODIE en los siguientes experimentos.

Con propósitos ilustrativos, la Ilustración 25 y la Ilustración 26 muestran las PC 2 y 8 para los modelos de Kurucz y las estrellas de ELODIE, respectivamente. Gráficamente aparecen razonablemente separadas en Kurucz, y de forma algo menos clara en ELODIE.

⁷ Nótese que el estudio de las temperaturas de los falsos positivos solo pudo hacerse con R, ya que Weka no ofrece esta flexibilidad en el tratamiento de los datos.

PC	FPR	FNR	Media Teff FP	Media Teff FN
2	0,1008	0,082	35228	8434
2,8	0,0672	0,01614	42385	8110
2,8,6	0,0672	0,01345	42385	8224
2,8,6,13	0,2184	0,0089	21874	8825
2,8,6,13,11	0,2268	0,0098	26205	8710
2,8,6,13,11,20	0,1932	0,0242	27367	7914
2,8,6,13,11,20,1	0,1932	0,0251	28390	7747
2,8,6,13,11,20,1,7	0,1680	0,0188	31506	7920
2,8,6,13,11,20,1,7,18	0,2857	0,0044	24433	8663

Tabla 6: Clasificación de estrellas frías (≤ 10013 K) y calientes (> 10013 K). Se muestra FPR, FNR, media de Teff en los falsos positivos y en los falsos negativos, en el rango azul, utilizando PCA y SVM como clasificador.

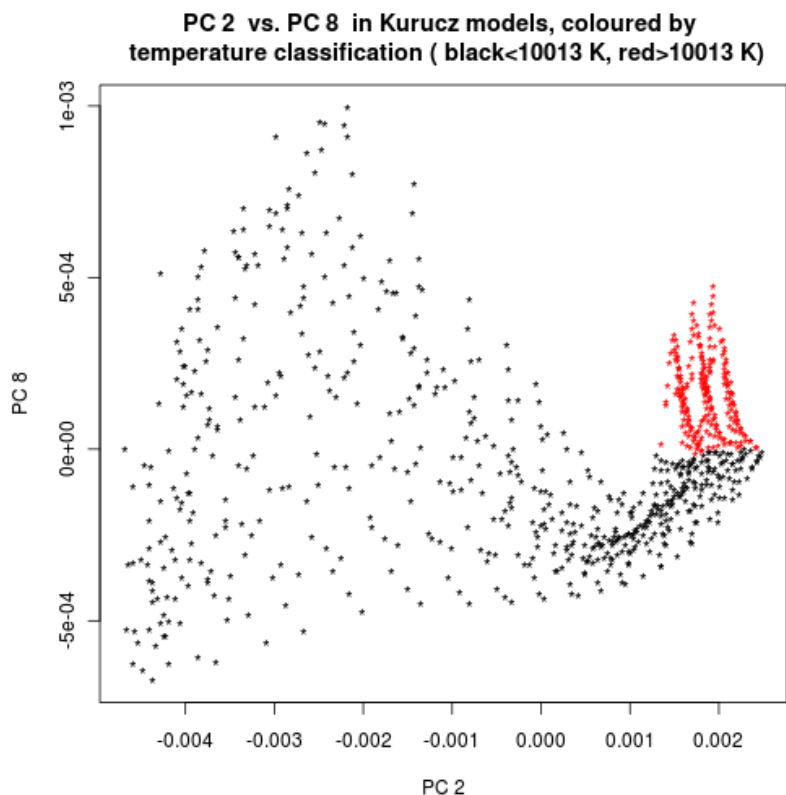


Ilustración 25: Componentes principales 2 y 8 en los modelos de Kurucz.

Aparecen en rojo las estrellas cuya Teff es mayor de 10013K, y en color negro las de menor temperatura.

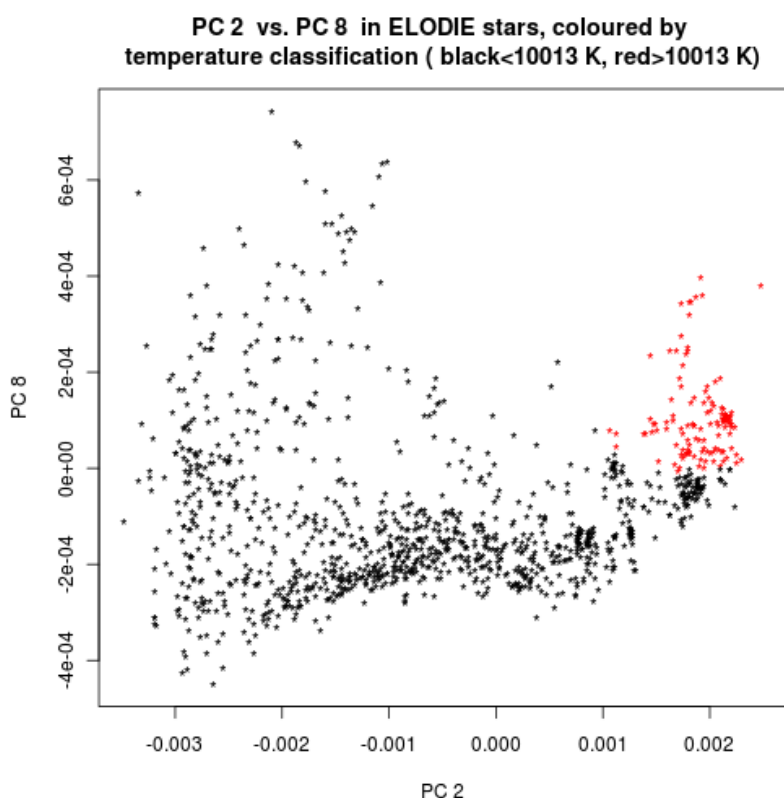


Ilustración 26: Componentes principales 2 y 8 en las estrellas de ELODIE.

En color rojo, las estrellas cuya T_{eff} es mayor de 10013K, y en color negro las de menor temperatura.

El hecho de que se haya conseguido un buen clasificador dicotómico a partir de una selección de las componentes principales pone de manifiesto un hecho que no puede ser pasado por alto: la acumulación de la varianza en los componentes principales consecutivos no es un buen criterio para seleccionar el número de componentes principales para obtener un clasificador dicotómico con SVM. De hecho, en la selección de las mejores componentes principales no se encuentra la primera componente principal, que es la componente que más varianza acumula de todas.

3.3.1.4 Experimentos finales de regresión con máquinas de vectores soporte (ϵ -SVM)

En los experimentos anteriores se descubrieron algunas peculiaridades de los conjuntos de datos que servirán en este apartado para mejorar los resultados:

- 1) Los modelos parecen ser mucho mejores cuando se restringe el conjunto de datos a las estrellas y modelos de estrellas cuya temperatura efectiva es menor de 10.000 K.
- 2) Es posible que una selección no consecutiva de las componentes principales produzca modelos de menor error que si se utilizan las $n(\sim 30)$ primeras componentes principales. En el clasificador dicotómico así ocurría.

Además, se había utilizado la validación cruzada de 10 subconjuntos para la selección de los mejores parámetros del kernel. Como alternativa, se entrenó el conjunto de datos de Kurucz validando el modelo con el conjunto de datos de ELODIE, es decir, eligiendo el valor de σ del kernel de Laplace que mejores resultados da con los datos de ELODIE.

En este apartado, se expondrán los experimentos realizados introduciendo los cambios anteriores, y comprobando cómo afectan a los modelos predictivos.

Las tablas 7 y 8 muestran los índices de error de un modelo entrenado con el conjunto completo de modelos de Kurucz, y otro modelo que ha sido entrenado con los modelos de Kurucz cuya temperatura se encuentra por debajo de 10000 K. En ambos casos, con validación cruzada de 10 subconjuntos y utilizando como conjunto de entrada las 30 primeras PC. Los conjuntos de test sobre los que se evalúa el modelo regresivo son los conjuntos de Kurucz y ELODIE. errores registrados son el índice de correlación (Corr), el error medio absoluto (MAE), y la raíz del error cuadrático medio (RMSE), cuyas definiciones se encuentran el apéndice I.

		KURUCZ			ELODIE		
		Corr	MAE	RMSE	Corr	MAE	RMSE
Rango azul	Teff	0,9977	91,68	634	0,8524	1232	2823
	log(g)	0,9991	0,0117	0,0661	0,6992	0,7121	0,8289
Rango rojo	Teff	0,9978	78,61	606,29	0,2301	3576	5360
	log(g)	0,9928	0,0461	0,1868	0,5625	0,7573	0,8782

Tabla 7: Resultados con 30 PC, validación cruzada, conjuntos completos.
 $\sigma=0,125$

Cuando se restringió el conjunto de estrellas a aquellas cuya temperatura efectiva es menor de 10.000K, el error en el conjunto de ELODIE disminuyó mucho en la temperatura efectiva, y empeoró ligeramente en el logaritmo de la gravedad. Este mismo efecto ocurre en Kurucz.

		KURUCZ			ELODIE		
		Corr	MAE	RMSE	Corr	MAE	RMSE
Rango azul	Teff	0,9981	16,79	97,93	0,788	341	463
	log(g)	0,9989	0,0148	0,0733	0,7799	0,8458	0,9561
Rango rojo	Teff	0,9969	20,51	124,68	0,6663	550	858
	log(g)	0,9906	0,0645	0,2225	0,5565	0,9545	1,08

Tabla 8: Resultados con 30 PC, validación cruzada, estrellas y modelos < 10000K.
 $\sigma=0,125$

El siguiente cambio fue utilizar una selección de componentes principales, en vez de utilizar la totalidad de las 30 primeras componentes principales. En el clasificador dicotómico un conjunto de 3 PC resultó ser mucho mejor que las 30 primeras PC. En consecuencia, parecía lógico pensar que una selección de PC podría mejorar los resultados. La selección de PC se llevó a cabo mediante la herramienta Weka, utilizando el algoritmo *WrapperSubsetEval* (17). Este método de búsqueda de mejores atributos utiliza a su vez un sub-algoritmo de inducción para la evaluación de atributos y un sub-algoritmo que selecciona los mejores atributos a partir de la evaluación. Ambos sub-algoritmos son elegibles de antemano, y *WrapperSubsetEval* los trata como cajas negras. El sub-algoritmo de selección de mejores atributos utiliza la evaluación del sub-algoritmo de evaluación para elegir el siguiente subconjunto de atributos que se evaluará en el siguiente paso. El algoritmo

termina cuando el sub-algoritmo de selección de atributos encuentra una selección óptima de atributos. El sub-algoritmo de evaluación elegido fue *SMOReg*, que es la implementación de la regresión mediante SVM en Weka. *BestFirst* fue el sub-algoritmo de selección de atributos elegido. *BestFirst* busca en el espacio de subconjuntos de atributos mediante una estrategia egoísta de selección, ampliada mediante una modificación : mantiene dos listas de atributos, una de mejores y otra de candidatos, y finaliza después de que los últimos k candidatos no hayan supuesto una mejora significativa en el aumento de la precisión.

Los resultados fueron los siguientes:

Rango azul, Teff: 1,2,3,4,6,7,8,13,14,15,16,17,18,21,26,27 (16 PC)

Rango azul, log(g): 2,5,6,7,10,11,12,14,15,17,19,20,25,29 (14 PC)

Rango rojo, Teff: 1,2,5,7,8,11,12,14,19,21 (10 PC)

Rango rojo, log(g): 1,2,3,4,7,8,10,13,14,15,17,18,21,22,23 (15 PC)

La Tabla 9 muestra los índices de error al utilizar solo los PC seleccionados de un modelo entrenado con los modelos de Kurucz cuya temperatura se encuentra por debajo de 10000 K, con validación cruzada.

		Corr	MAE	RMSE	Sesgo	Desv. sesgo
Rango azul Kurucz	Teff	0,9981	18	99	-15	98
	log(g)	0,9977	0,0295	0,1109	-0,0059	0,1108
Rango rojo Kurucz	Teff	0,9942	31	170	-25	168
	log(g)	0,9951	0,0443	0,1616	0,0135	0,1612
Rango azul ELODIE	Teff	0,931	269	384	-84	375
	log(g)	0,7863	0,6736	0,8611	-0,5892	0,6283
Rango rojo ELODIE	Teff	0,8868	396	544	267	474
	log(g)	0,5677	0,7038	0,8926	-0,4417	0,776
<p><i>Tabla 9:</i> Resultados con PC seleccionados , validación cruzada, estrellas y modelos < 10000K. $\sigma = (0.25, 0.125, 0.25, 0.25)$</p>						

Utilizando una selección de componentes principales, se reducen notablemente los errores de log(g) en ambos rangos de longitud de onda, y de Teff en el rango rojo. Los errores en Kurucz aumentan o disminuyen solo ligeramente.

El siguiente cambio consistió en utilizar el conjunto de datos de ELODIE como conjunto de validación, en vez de la validación cruzada. La validación ELODIE buscó el valor de σ del kernel que mejores resultados ofrecía, aunque el entrenamiento siguió haciéndose con los modelos de Kurucz. La Tabla 10 muestra los resultados de la ejecución, incluyendo el sesgo y la desviación típica de los errores.

		Corr	MAE	RMSE	Sesgo	Desv. sesgo
Rango azul Kurucz	Teff	0,9947	38	170	-33	166
	log(g)	0,9928	0,0865	0,2005	0,0065	0,2005
Rango rojo Kurucz	Teff	0,9863	88	269	-47	265
	log(g)	0,9729	0,168	0,3867	0,0322	0,3856
Rango azul ELODIE	Teff	0,9393	236	352	-30	351
	log(g)	0,8113	0,6318	0,8175	-0,5469	0,6079
Rango rojo ELODIE	Teff	0,8934	359	512	-155	488
	log(g)	0,6337	0,6332	0,8052	-0,373	0,7139

Tabla 10: Resultados con PC seleccionados , validación ELODIE, estrellas y modelos <10000K.
 $\sigma=0,0625$

El mejor valor de σ resultó ser 0,0625 en todos los casos, y como cabía esperar, los errores en Kurucz empeoraron, mientras que en ELODIE mejoraron ligeramente.

En conclusión:

- En el rango azul de longitudes de onda, los modelos generados producen menor error para Teff.
- Restringir el conjunto de estrellas a aquéllas cuya temperatura está por debajo de los 10.000 K hace que el modelo de predicción de la temperatura efectiva mejore notablemente en ambos rangos de longitud de onda.
- Si se entrenan los modelos predictivos con una selección de PC, los errores del logaritmo de la gravedad en ambos rangos y la temperatura efectiva en el rango rojo mejoran notablemente.
- La validación de los modelos predictivos con el conjunto de ELODIE da como resultado que cambie el valor de σ a un valor menor y mejora ligeramente los resultados en ELODIE.

Las ilustraciones siguientes muestran los valores predichos frente a los valores reales del último experimento, para los parámetros físicos (temperatura efectiva y logaritmo de la gravedad) y los dos rangos de longitud de onda. En la Ilustración 27 se muestra la temperatura predicha frente a la real en el rango azul. Las malas predicciones por debajo de los 4000 K son hasta cierto punto lógicas, pues los modelos de Kurucz (el conjunto de entrenamiento) carecen de ejemplos por debajo de esa temperatura. Por encima de los 8000 K, algunas de las predicciones superan los 1000 K. Las mismas consideraciones se pueden hacer para las predicciones en el rango rojo, véase la Ilustración 29. La Ilustración 28 muestra la gravedad predicha frente a la real en el rango azul. Se aprecia que la mayoría de las estrellas de ELODIE están en torno a 4, que es donde se acercan más a su valor real. Por debajo de 3, las predicciones parecen tener un sesgo negativo. Las predicciones en la Ilustración 30, correspondiente a la gravedad en el rango rojo, parecen ser peores que sus correspondientes en el rango azul, especialmente por debajo de 3.

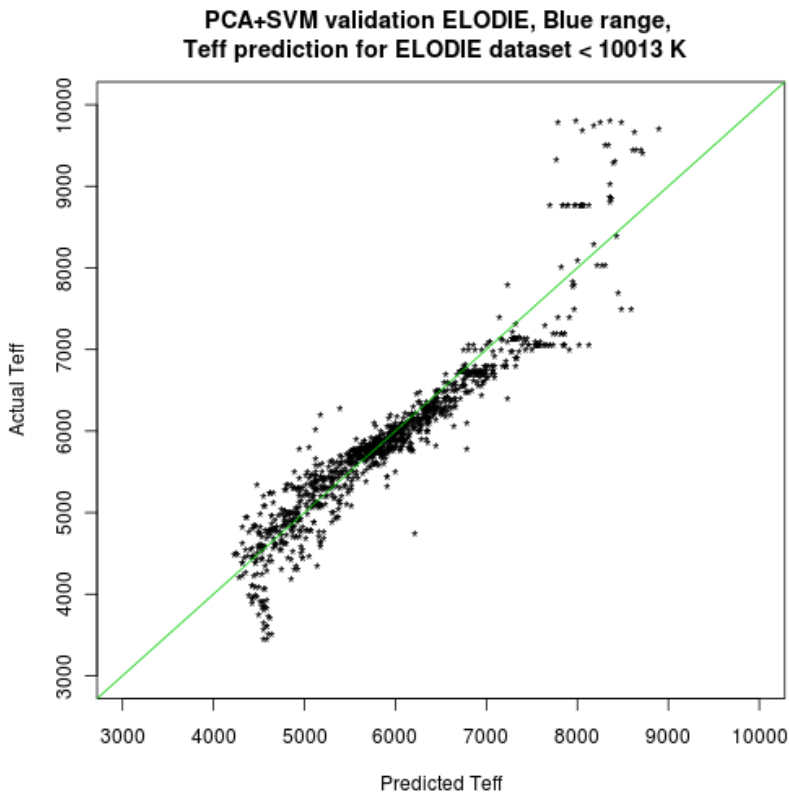


Ilustración 27: Temperatura efectiva predicha vs. real para estrellas por debajo de 10.000 K, rango azul, utilizando PCA + SVM

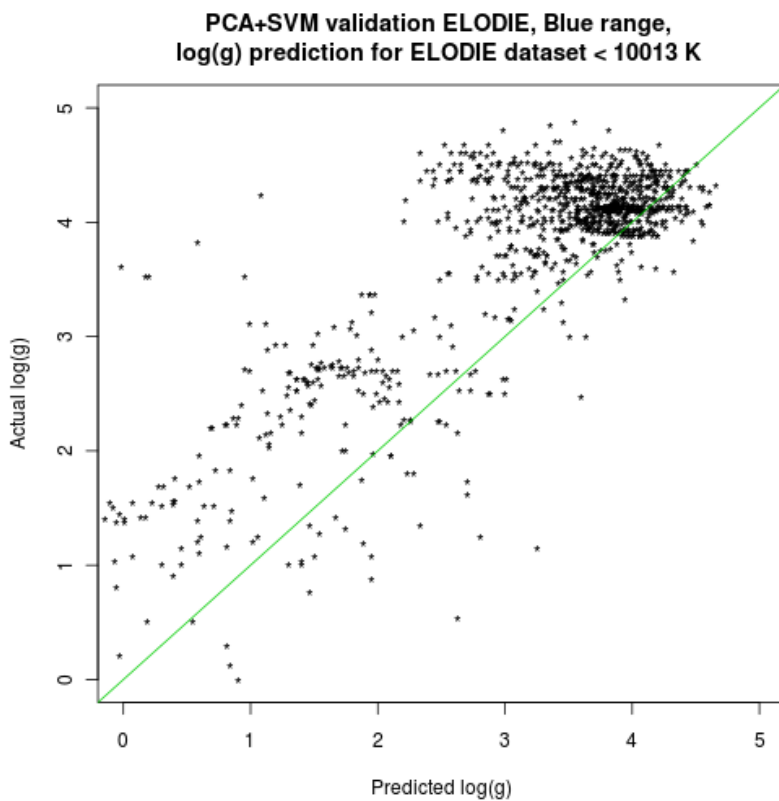


Ilustración 28: Gravedad predicha vs. real para estrellas por debajo de 10.000 K, rango azul, utilizando PCA + SVM

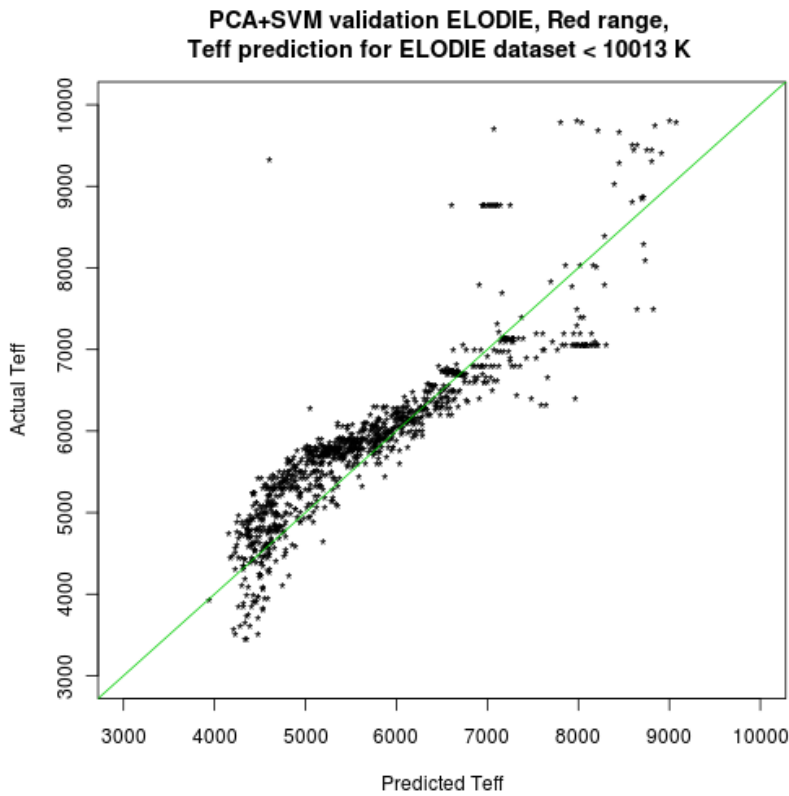


Ilustración 29: Temperatura efectiva predicha vs. real para estrellas por debajo de 10.000 K, rango rojo, utilizando PCA + SVM

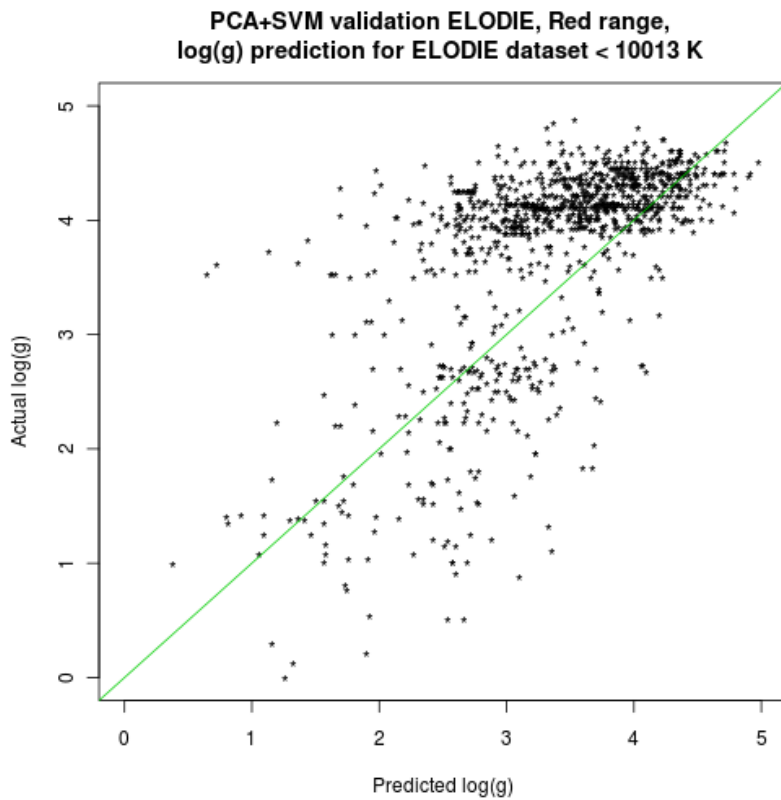


Ilustración 30: Gravedad predicha vs. real para estrellas por debajo de 10.000 K, rango rojo, utilizando PCA + SVM

3.3.2 Aplicación de los modelos predictivos a FLAMES

Una vez obtenidos unos modelos aceptables para las estrellas menores de 10000K (estrellas frías), el siguiente paso era aplicarlos a los espectros de FLAMES I y II. Como dichos conjuntos de datos no están clasificados, fue necesario clasificar primero las estrellas entre estrellas mayores y menores de 10000K, utilizando el clasificador expuesto en el apartado 3.3.1.3. El clasificador se aplicaba solamente sobre el rango azul de longitudes de onda, por lo que hubo que aplicarlo a las estrellas de FLAMES I y II del rango azul, y luego seleccionar las estrellas del rango rojo que tuviesen como identificador alguno de los del grupo seleccionado en el rango azul. La siguiente tabla muestra el número de estrellas clasificadas como estrellas frías y su porcentaje sobre el total de estrellas, para cada rango de longitudes de onda y conjunto de datos:

	Rango azul	Rango rojo
FLAMES I	2855 (84%)	5805 (64%)
FLAMES II	5589 (86%)	12195 (84%)

Sobre los conjuntos de datos seleccionados se aplicaron los modelos predictivos, y de tales predicciones, no se puede calcular el porcentaje de acierto, si bien se pueden representar de forma conjunta los dos parámetros predichos (T_{eff} y $\log(g)$) y valorar la forma en que se distribuyen. La Ilustración 31 muestra dicha relación para FLAMES I, y la Ilustración 32 la muestra para FLAMES II.

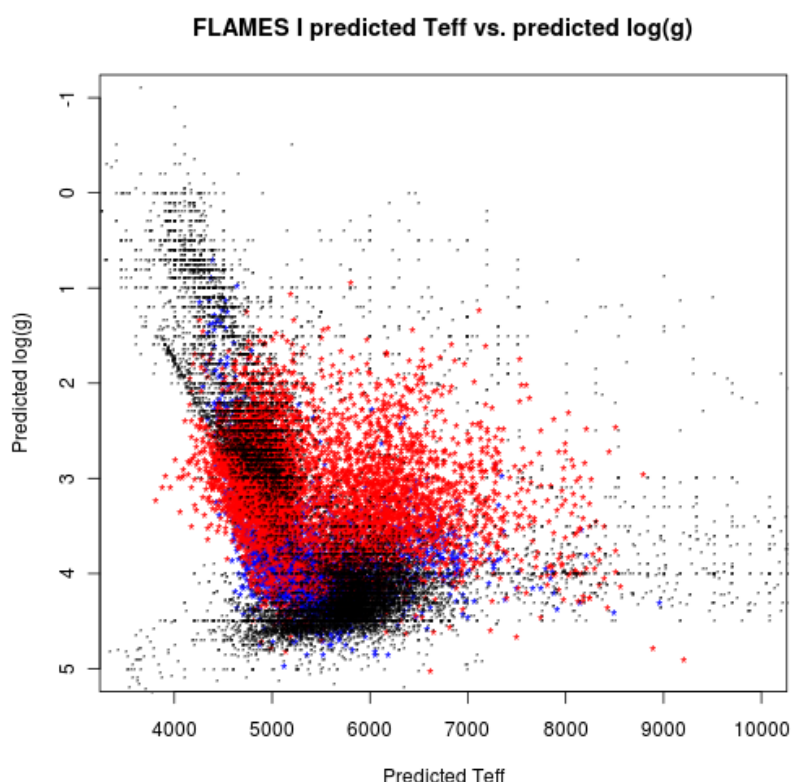


Ilustración 31: T_{eff} y $\log(g)$ predichos para FLAMES I.

En rojo, se muestran las predicciones para estrellas cuyo rango de longitudes de onda es el rojo, y en azul las estrellas predichas en el rango azul. Los puntos en color negro corresponden a la base de datos B/PASTEL

En ambas gráficas, se muestran las predicciones del rango de longitudes de onda azul en color azul, y las del rango rojo en rojo. Además, se han incluido las temperaturas y gravedades reales del conjunto de datos B/PASTEL (10), que se muestran en color negro. B/PASTEL es un conjunto de 15.495 estrellas reales cuyas temperaturas y gravedades que han sido recopiladas de diversas

fuentes y cuya veracidad se da por cierta.

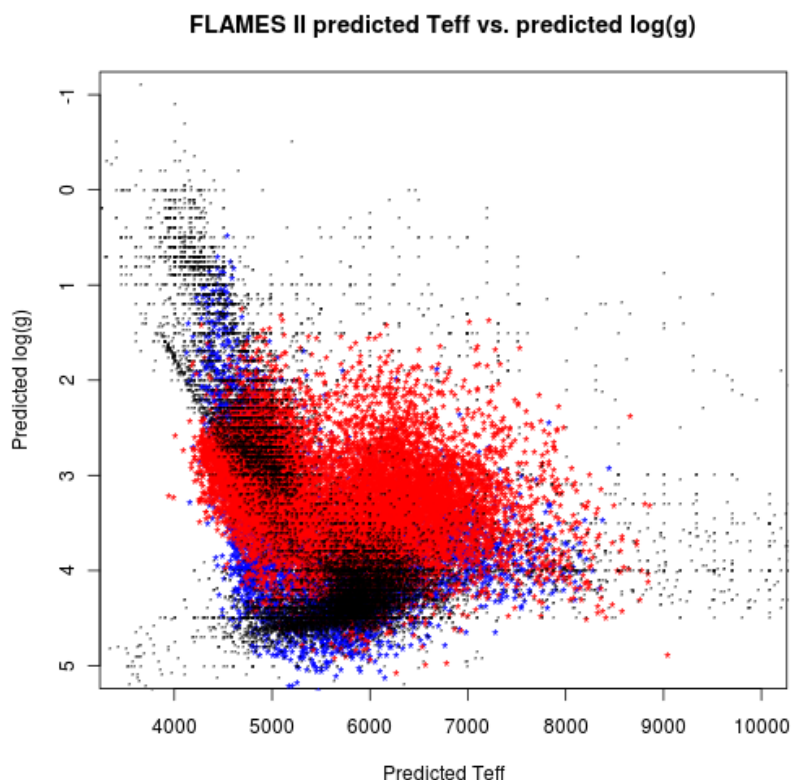


Ilustración 32: T_{eff} y $\log(g)$ predichos para FLAMES II.

En rojo, se muestran las predicciones para estrellas cuyo rango de longitudes de onda es el rojo, y en azul las estrellas predichas en el rango azul. Los puntos en color negro corresponden a la base de datos B/PASTEL.

Las dos ilustraciones anteriores muestran como el rango azul se acerca más a la distribución de temperaturas y gravedades de B/PASTEL.

4 Otros experimentos

4.1 Fusión de espectros en FLAMES

Los conjuntos de datos FLAMES I y II vienen identificados con una cadena en la que se expresa la estrella de donde proviene el espectro y el momento en el que se tomaron los datos, y en el rango rojo de longitudes de onda, de cada estrella puede haber entre 1 y 4 espectros diferentes. Este hecho originó la idea de fundir varios espectros en un solo para cada estrella que tuviera más de un espectro, con objeto de mejorar la relación señal-ruido, ya que de este modo se eliminaban rayos cósmicos y se obtenía el valor medio de flujo en cada longitud de onda, lo cual podría suponer disminuir la varianza y eliminar ruido del espectro.

Para construir el nuevo espectro fundido, se eligió la mediana del flujo por cada longitud de onda para espectros de la misma estrella. Al aplicar los modelos predictivos sobre el conjunto fundido de estrellas, no se observaron diferencias significativas en los parámetros predichos, que tenían una distribución muy parecida a los espectros originales.

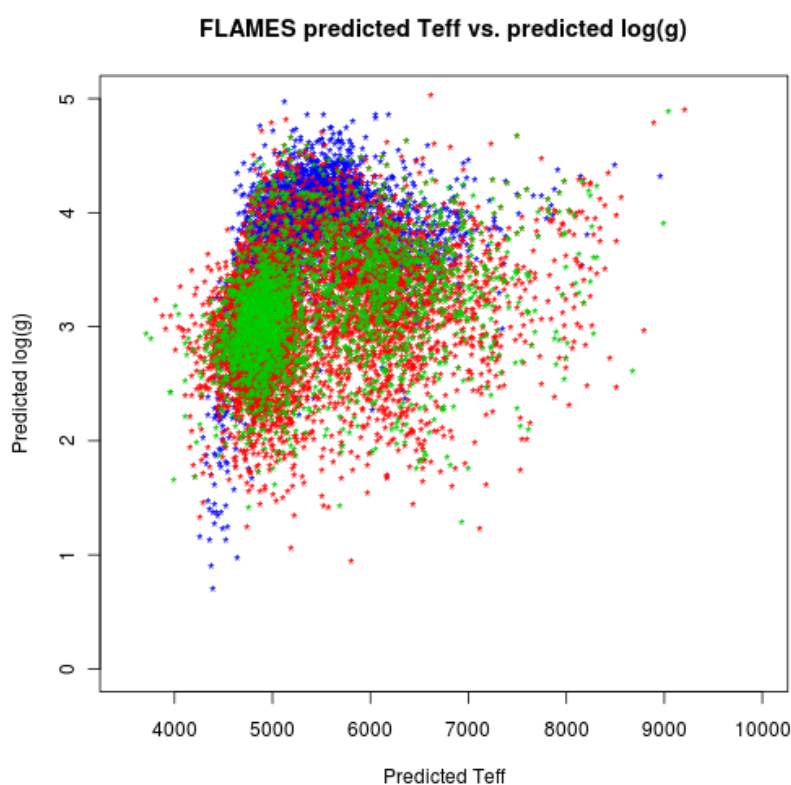


Ilustración 33: Temperatura efectiva predicha frente a gravedad predicha en FLAMES I, para el rango azul (en color azul), en rango rojo (en color rojo) y las estrellas fundidas en el rango rojo (en color verde). Se utilizó PCA + SVM para la predicción.

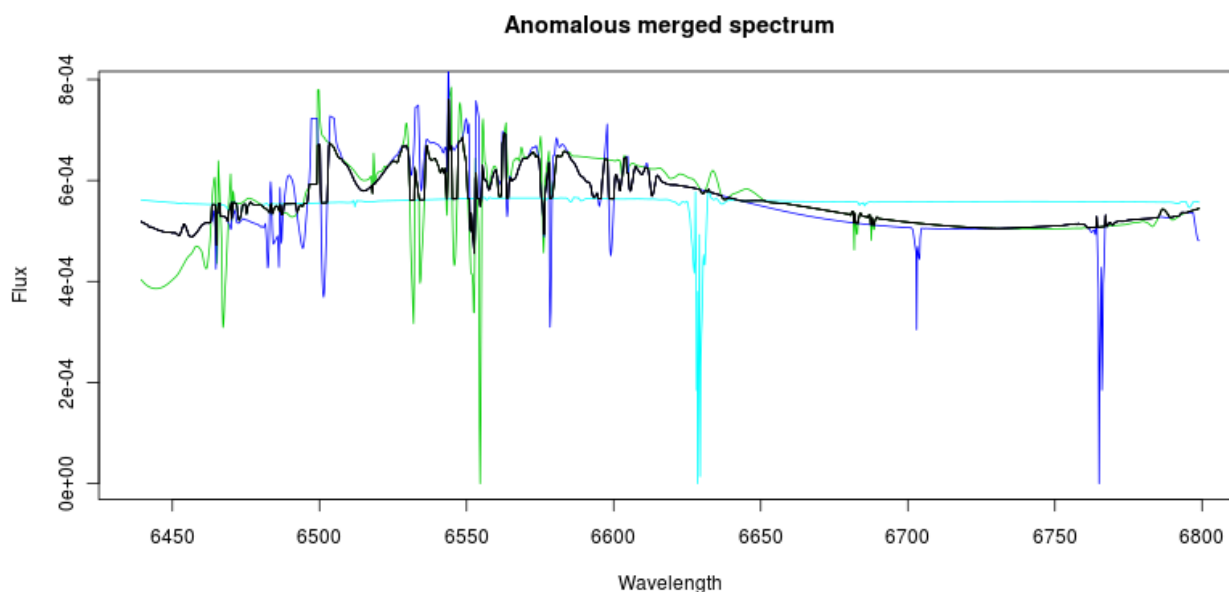


Ilustración 34: Espectro fundido anómalo en FLAMES I en color negro, y los espectros originales de donde proviene en otros colores, todos de la misma estrella: 100740166

En la Ilustración 33 se muestran la temperatura efectiva predicha frente al logaritmo de la gravedad predicha para el conjunto de espectros en el rango azul, en el rango rojo y para el conjunto de espectros fundidos en el rango rojo, estos últimos representados en color verde. El modelo predictivo fue PCA y SVM para estrellas menores de 10013 K (apartado 3.3.1.4). El conjunto de

espectros fundidos en rango rojo ocupan aproximadamente el mismo espacio que los espectros sin fundir en el rango rojo. No se tienen pruebas para afirmar que los espectros fundidos son mejores o peores que los espectros sin fundir. Es conveniente recordar que FLAMES es un conjunto de espectros sin clasificar, y es difícil estimar si el cambio supone una mejora en su predicción.

Además, en algunos espectros fundidos se observaron anomalías, zonas en las que el valor de flujo parecía tener un límite superior o inferior, con un aspecto escalonado, algo que no es natural en los espectros reales. El motivo de estos espectros anómalos es que se encontraron espectros de la misma estrella muy diferentes entre sí. En la Ilustración 34 se muestra en color negro un espectro fundido y los tres espectros originales de donde proviene, de diversos colores. Las razones de tales diferencias escapan al propósito del presente estudio. Se estima que el número de estos espectros anómalos es bastante escaso.

Debido a las anteriores razones, se decidió abandonar la fusión de espectros y el conjunto de datos resultante, no tanto por suponer que fuera una técnica fallida, sino por la creencia de que su utilización no iba a suponer una mejora significativa en las predicciones.

4.2 K-vecinos más próximos

En el método de los K-vecinos más próximos ponderados (16) se determinan las k instancias del conjunto de entrenamiento más cercanas al ejemplo que se quiere predecir. La media ponderada de las k instancias es el valor predicho, teniendo tanto más peso la instancia cuanto más próxima se encuentre al ejemplo. El algoritmo es el siguiente:

1. Sea $L = \{(\mathbf{x}_j, \mathbf{y}_j), j = 1..n\}$ un conjunto de n observaciones \mathbf{x}_j (espectros), cada una asociada a un atributo de salida \mathbf{y}_j (temperatura o gravedad). Sea x una nueva observación cuya clase y debe ser estimada.
2. Encontrar los $k + 1$ vecinos más próximos de acuerdo a la medida de distancia $d(x, x_i)$. La medida de distancia no tiene por qué ser la distancia euclídea. Por ejemplo, se puede utilizar la distancia de Minkowsky variando el parámetro p :

$$d_{Minkowsky}(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^D |x_i - y_i|^p \right)^{1/p} \quad \text{donde } D \text{ es el número de dimensiones}$$

3. La distancia del vecino cuya distancia es mayor (vecino $(k + 1)$ -ésimo) se utiliza para estandarizar el resto de vecinos más próximos, mediante la fórmula:

$$D_i = D(x, x_i) = \frac{d(x, x_i)}{d(x, x_{k+1})} \quad \text{para } 1 \leq i \leq k$$

4. Transformar las distancias normalizadas (D_i) en una medida de similitud (w_i) a través de una función (K), cuyo resultado sea un peso que sea tanto mayor cuanto menor sea la distancia: $w_i = K(D_i)$ Esta función determina la influencia que tiene cada nodo vecino, que es inversa a su distancia. Estos son algunos ejemplos:

$$\text{Triangular: } K(d) = (1 - d)$$

$$\text{Inversa: } K(d) = \frac{1}{d}$$

$$\text{Gaussiana: } \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{d^2}{2}\right)$$

5. La estimación final (\hat{y}) es la media ponderada de los k vecinos más próximos:

$$\hat{y} = \frac{\sum_{i=1}^k w_i y_i}{\sum_{i=1}^k w_i}$$

Esta técnica puede ser ajustada modificando la función de transformación K , y el número de vecinos k . El paquete utilizado en R para llevar a cabo las pruebas con esta técnica fue *kknn*. Se utilizó la distancia euclídea como medida de distancia entre las estrellas, la transformación triangular, y el número de vecinos (k) igual a 7.

Los resultados de aplicar esta técnica sobre el conjunto completo de estrellas se muestran en la Tabla 11, y sobre la selección de estrellas y modelos de estrellas menores de 10.000 K en la Tabla 12.

		Corr	MAE	RMSE	μ_e	σ_e
Rango azul	Teff	0.8393	845	2992	137	2990
	log(g)	0.8399	0,5865	0,7924	-0,064	0,7901
Rango rojo	Teff	0,7332	1278	3864	-65	3865
	log(g)	0,6364	0,6519	0,8671	-0,2504	0,8305
<i>Tabla 11: Resultados sobre conjunto de test ELODIE, utilizando 7 vecinos más próximos en Kurucz, para los conjuntos completos de estrellas.</i>						

		Corr	MAE	RMSE	μ_e	σ_e
Rango azul	Teff	0,9262	420	506	225	453
	log(g)	0,8562	0,6044	0,8165	-0,1172	0,8084
Rango rojo	Teff	0,9079	369	559	260	495
	log(g)	0,77	0,5192	0,6703	-0,2435	0,6248
<i>Tabla 12: Resultados sobre conjunto de test ELODIE, utilizando 7 vecinos más próximos en Kurucz, para estrellas y modelos < 10013 K.</i>						

Como en el caso de PCA y SVM, la técnica de los 7 vecinos más próximos mejora notablemente la predicción de la temperatura efectiva cuando se restringe a las estrellas y modelos de estrellas cuya temperatura efectiva es menor de 10.000 K, si bien los resultados son ligeramente peores que en PCA y SVM (Tabla 10). La predicción del logaritmo de la gravedad no mejora en el rango azul de longitudes de onda, pero en el rango rojo, se aprecia una gran reducción de RMSE (0,6703), siendo mejor que el obtenido para PCA + SVM (RMSE=0,8052), e igualando a las de PLS (RMSE=0,6627), que son las mejores predicciones hechas en la investigación, con la única

diferencia del sesgo, que en PLS es claramente menor ($\mu_e = -0,025$)

Las ilustraciones 35 a 38 muestran los valores predichos frente a los valores reales del último experimento, tanto para la temperatura efectiva como para el logaritmo de la gravedad, y en los dos rangos de longitud de onda.

La temperatura en el rango azul presenta un ligero sesgo positivo entre los 5000K y 7000K, y la predicción en el rango 4000K a 5000K es muy mala, el modelo parece asignar casi siempre valores cercanos a 4000K. A partir de los 7000K las predicciones empeoran progresivamente, aunque también desciende el número de estrellas en ELODIE en ese rango. En el rango rojo, la predicción de la temperatura tiene un menor error absoluto medio y un mayor error cuadrático medio. En este rango no presenta la anomalía en el rango de los 4000K a 5000 K, pero a partir de los 7000 K las predicciones son malas, llegando a errores absolutos superiores a los 2000 K. Esta última circunstancia puede ser la causante de que el RMSE sea menor en el rango azul, a pesar de que el aspecto de la gráfica en el rojo parezca mejor.

En el entorno de 4, la gravedad presenta un sesgo positivo en el rango azul, y negativo en el rojo. La mayoría de las estrellas se encuentran en el rango [4,5]. Las que tienen gravedad menor que 4 se predicen mejor en el rango rojo.

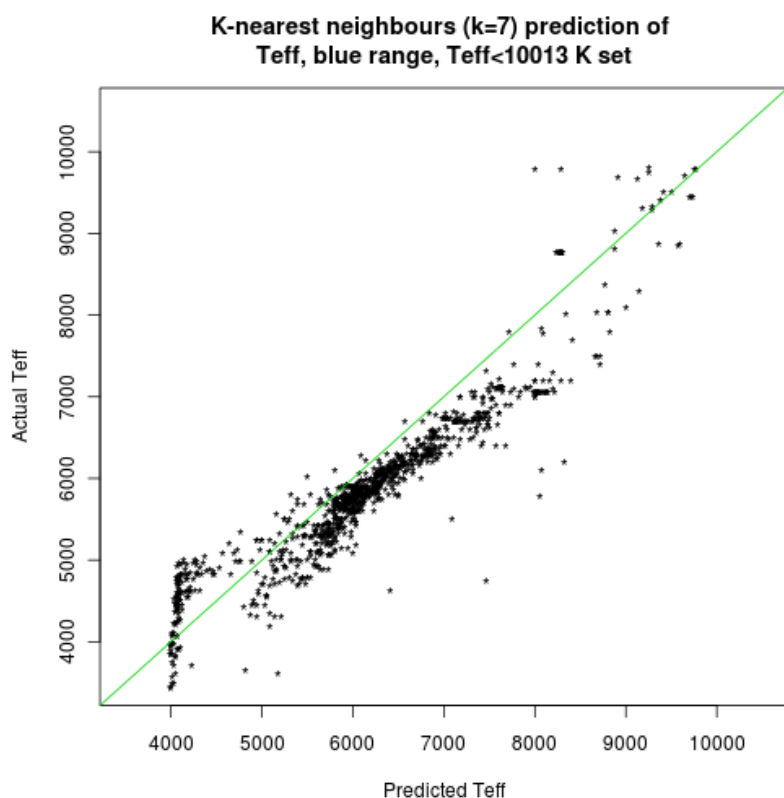


Ilustración 35: Temperatura efectiva predicha vs. real para estrellas por debajo de 10.000 K, rango azul, utilizando 7 vecinos más próximos.

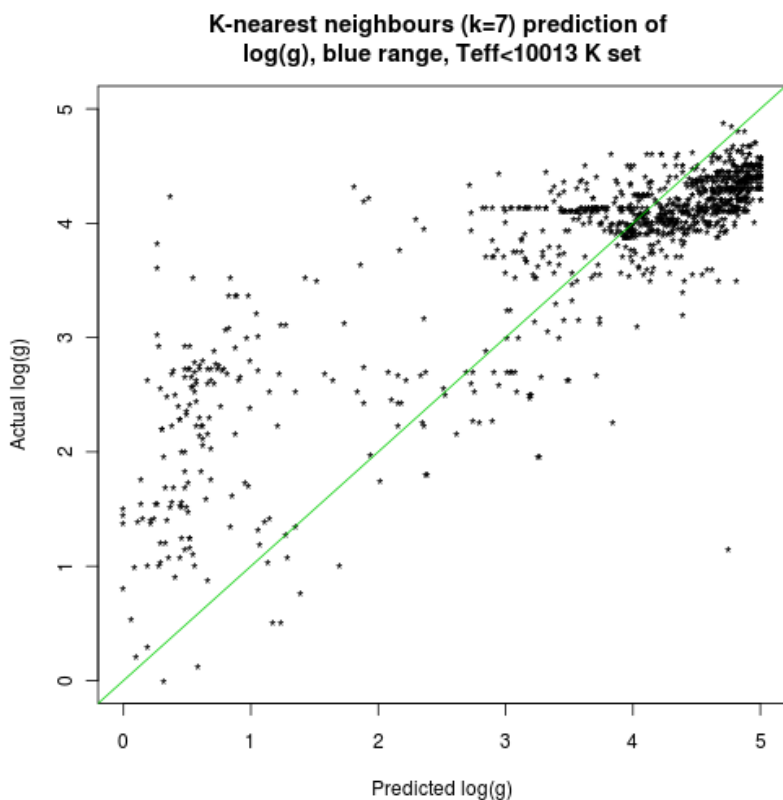


Ilustración 36: Gravedad predicha vs. real para estrellas por debajo de 10.000 K, rango azul, utilizando 7 vecinos más próximos

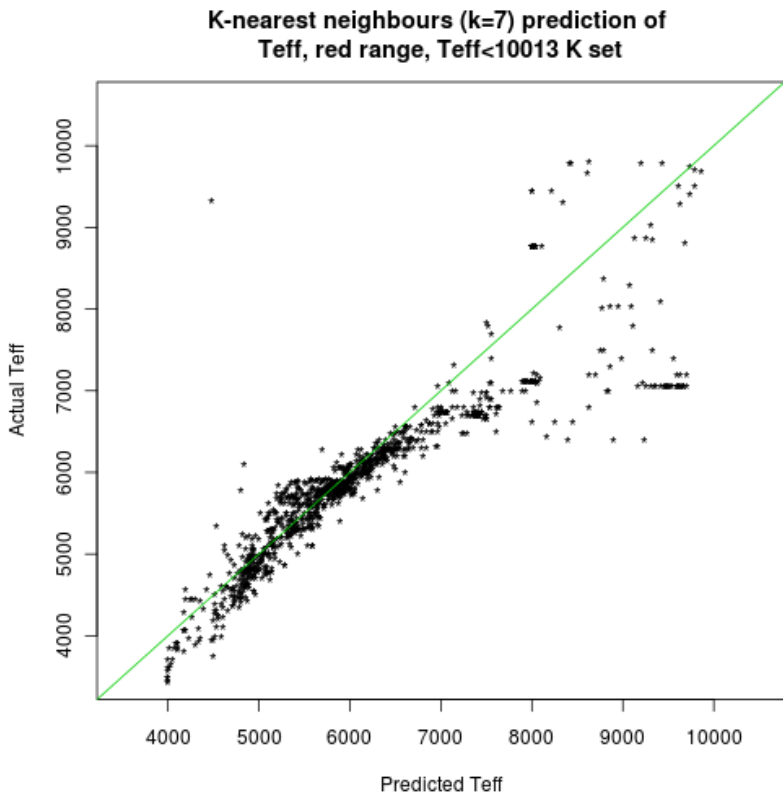


Ilustración 37: Temperatura efectiva predicha vs. real para estrellas por debajo de 10.000 K, rango rojo, utilizando 7 vecinos más próximos

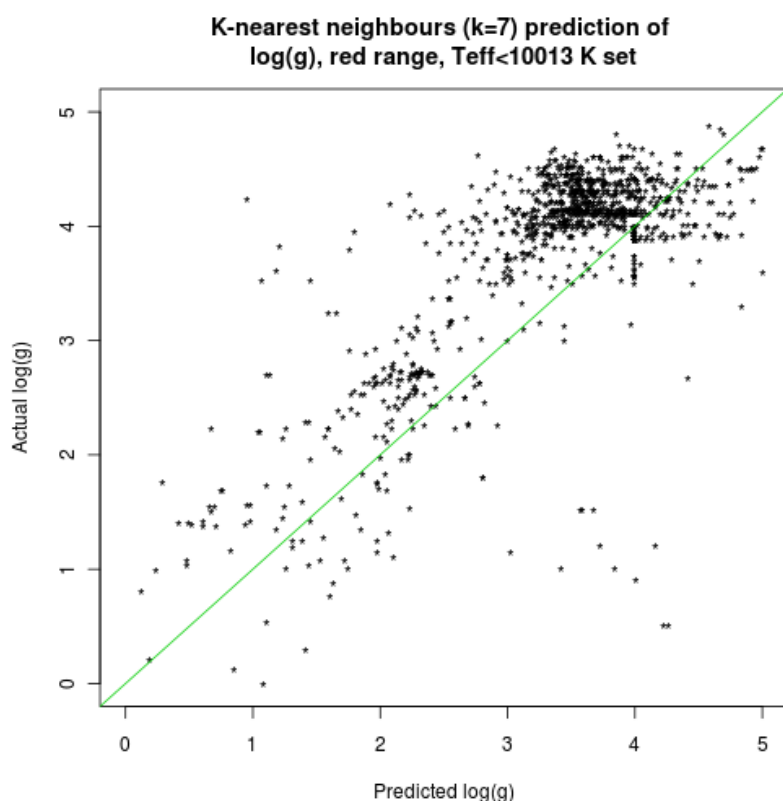


Ilustración 38: Gravedad predicha vs. real para estrellas por debajo de 10.000 K, rango rojo, utilizando 7 vecinos más próximos

En un intento de mejorar los resultados, se utilizó la función *kknn.train* para hallar los parámetros del algoritmo que menor error tienen, mediante la técnica *leave-one-out*, que consiste en extraer cada uno de los espectros del conjunto de datos y tratar de predecir el parámetro físico en cuestión a partir del resto de los espectros. Esta función utiliza el error cuadrático medio para medir la bondad del clasificador, y repite el proceso de evaluación con cada combinación de parámetros del algoritmo (número de vecinos, función de transformación). La combinación que menor error ofrezca es la que finalmente se elige. Las funciones de transformación utilizadas fueron la rectangular, triangular, *epanechnikov*, *biweight*, *triweight*, coseno, inversa, y gaussiana (véase (16) para conocer todas las funciones). El número de vecinos varió de 1 a 11. El RMSE de la evaluación *leave-one-out* sobre el propio conjunto de Kurucz, para estimar la temperatura efectiva en estrellas menores de 10000K, fue de 232 K y 299 K en los rango azul y rojo, respectivamente. La estimación del logaritmo de la gravedad fue de 0,36 y 0,49 en los rangos azul y rojo, respectivamente.

Sin embargo, al evaluar los modelos sobre el conjunto de ELODIE, los resultados fueron mucho peores que los mostrados en la Tabla 12. La causa de estos malos resultados pudiera deberse a un problema de sobreajuste, quizá debido al método *leave-one-out*, considerado peor método de evaluación que otros más utilizados como la validación cruzada, o *bootstrapping* (véase (13) páginas 434-466).

4.3 Diffusion maps y SVM

PCA es una técnica de reducción de dimensionalidad que lleva aparejada la asunción de linealidad, ya que los datos originales son proyectados en un hiperplano con un número de dimensiones mucho menor. Con objeto de buscar un método alternativo de reducción de dimensionalidad no lineal, se experimentó con los *diffusion maps* (4). Esta técnica trata de obtener una representación alternativa

(*las coordenadas de difusión*) de los datos originales . En vez de utilizar el concepto de distancia clásica entre dos vectores para expresar su similitud entre ellos, se utiliza en concepto de paseo aleatorio de Markov (*random walk*) entre un vector y otro. Supongamos que se asocia una probabilidad entre dos vectores, de tal forma que sea tanto mayor cuanto menor distancia euclídea haya entre los dos. Esta probabilidad se puede asociar a la transición entre un vector y otro, o dicho de otro modo, a que un caminante imaginario dé un paso entre dichos vectores. Si se puede calcular una segunda probabilidad: la de que, desde un vector x , en una secuencia de t transiciones de un vector a otro (un paseo de t pasos), se alcance al vector y . A esta segunda probabilidad se le denomina *distancia de difusión*. Según esta distancia, dos vectores están tanto más cerca cuanto más fácil sea ir de un vector a otro dando pasos cortos. Esta segunda distancia es la utilizada para construir las *coordenadas de difusión*.

El punto de partida del algoritmo es un grafo ponderado, donde los nodos son los vectores de entrada (los espectros) . El peso w de cada arista viene dado por la fórmula:

$$w(x, y) = \exp\left(-\frac{d(x, y)^2}{\epsilon}\right)$$

donde $d(x, y)$ es la medida de distancia entre los vectores x e y

ϵ es un parámetro de ajuste. Debe ser lo suficientemente pequeño para que el peso sea cercano a 0 a no ser que x e y sean similares, pero lo suficientemente grande para que el grafo construido sea conexo.

A partir de estos pesos, se construye el paseo aleatorio de Markov sobre dicho grafo. Para ello se asocia una probabilidad entre cada nodo cuyo valor se define como:

$$p_1(x, y) = \frac{w(x, y)}{\sum_z w(x, z)} \quad \text{donde } z \text{ es el conjunto de vectores considerado}$$

La función p_1 representa la probabilidad de moverse del vector x al vector y en 1 paso. Esta probabilidad es cercana a cero, a no ser que x e y sean similares. Si el número de vectores es n , se denotará por P a la matriz $n \times n$ que contiene los valores p_1 . De la teoría de cadenas de Markov se conoce que la potencia P^t contiene las probabilidades $p_t(x, y)$ de moverse de x a y en t pasos. La distancia de difusión D_t se define como:

$$D_t^2(x, y) = \sum_z \frac{(p_t(x, z) - p_t(y, z))^2}{\phi_0(z)}$$

donde $\phi_0(z)$ es la distribución estacionaria del paseo aleatorio en z , es decir, la probabilidad a largo plazo de que en un paseo aleatorio el caminante se encuentre en el nodo z . La distancia $D_t(x, y)$ será pequeña solo si x e y están conectados por muchos caminos con peso alto.

El paso final consiste en encontrar una representación de los vectores que refleje las distancias de difusión entre ellos como distancias euclídeas. La descomposición espectral biortogonal de la matriz P^t es:

$$p_t(x, y) = \sum_{j \geq 0} \lambda_j^t \psi_j(x) \phi_j(y)$$

donde λ son los valores propios de P

ψ son los vectores propios de P por la izquierda

φ son los vectores propios de P por la derecha

Según lo anterior, se demuestra que la distancia de difusión puede ser definida como:

$$D_t^2(\mathbf{x}, \mathbf{y}) = \sum_{j \geq 1} \lambda_t^{2t} (\psi_j(\mathbf{x}) + \psi_j(\mathbf{y}))^2$$

Se define el mapa de difusión Ψ de los m primeros autovalores como:

$$\Psi_t : \mathbf{x} \rightarrow [\lambda_1^t \psi_1(\mathbf{x}), \lambda_2^t \psi_2(\mathbf{x}), \dots, \lambda_m^t \psi_m(\mathbf{x})] \quad \text{de } \mathbb{R}^p \text{ a } \mathbb{R}^m \quad \text{con } p \gg m$$

Finalmente, una buena aproximación a la distancia de difusión se obtiene con los m primeros vectores propios por la izquierda:

$$D_t^2(\mathbf{x}, \mathbf{y}) \approx \sum_{j=1}^m \lambda_t^{2t} (\psi_j(\mathbf{x}) + \psi_j(\mathbf{y}))^2 = \|\Psi_t(\mathbf{x}) - \Psi_t(\mathbf{y})\|^2$$

El mapa de difusión establece una correspondencia entre los vectores de entrada, de dimensión p , a vectores de dimensión m mucho menor, consiguiendo de esta forma una reducción de la dimensionalidad y un conjunto de vectores cuya distancia euclídea entre ellos es aproximadamente la distancia de difusión entre los vectores originales.

En la herramienta R, el paquete *diffusionMap* contiene una implementación del algoritmo para calcular las coordenadas de difusión.

De la forma de hallar el peso w del grafo inicial, se deduce que los *diffusion maps* pueden ser ajustados de varias formas para tratar de mejorar los resultados más adelante con SVM: variando la medida de distancia y el valor de ϵ . Para encontrar la mejor configuración de las coordenadas de difusión, se realizaron pruebas con variaciones de p en la distancia de Minkowsky, que se define como:

$$d_M(x, y) = \left(\sum_{i=1}^n (|x_i - y_i|)^p \right)^{1/p}$$

La 30 primeras coordenadas de difusión con distintos valores de p fueron utilizadas con regresión SVM, y aunque los errores obtenidos fueron en general malos, $p=6$ parecía ser el valor p de la distancia Minkowsky en las que más se reducía el error. Para obtener el mejor valor de ϵ , se diseñó una función en R que calculaba el RMSE variando ϵ sucesivamente.

En los primeros experimentos, se calcularon las 30 primeras coordenadas de difusión entre un conjunto de datos de 1.660 estrellas formado por una mezcla entre los modelos de Kurucz y las estrellas de ELODIE, a partes iguales. Es necesario hallar las coordenadas de difusión uniendo el conjunto de entrenamiento (Kurucz) y el de validación (ELODIE), debido a que las coordenadas de difusión de cada vector de entrada se calculan a partir de todos los valores y vectores propios de la matriz P^t , que a su vez se calcula a partir de todas las distancias entre los vectores de entrada. Un cambio en la matriz P^t puede alterar todos los valores y vectores propios. Nótese la diferencia con respecto a PCA, donde las nuevas coordenadas se obtienen a partir de una combinación lineal que depende solamente del vector de entrada considerado, y no de todo el conjunto de vectores de entrada. De hecho, para calcular el mapa de difusión solo se necesitan las distancias entre los vectores, y no los vectores en sí.

Las ilustraciones 39 y 40 muestran las tres primeras coordenadas de difusión obtenidas en ambos conjuntos.

Diffusion coordinates in Kurucz (black) and ELODIE (red)

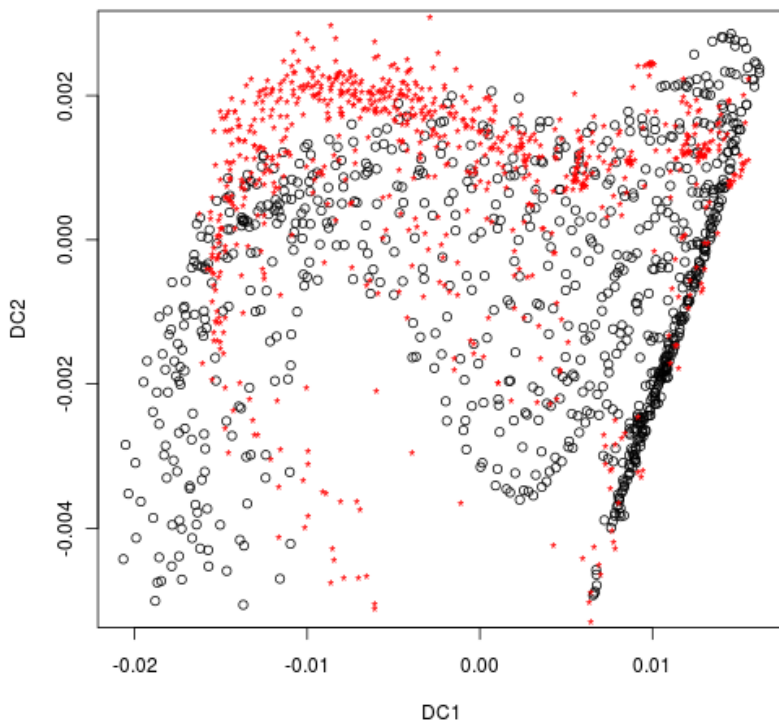


Ilustración 39: Coordenadas de difusión 1 y 2.

Los modelos de Kurucz aparecen en color negro, y los de ELODIE, en rojo. Muchos de los puntos del conjunto de ELODIE se salen del espacio que ocupan los puntos de Kurucz.

Diffusion coordinates in Kurucz (black) and ELODIE (red)

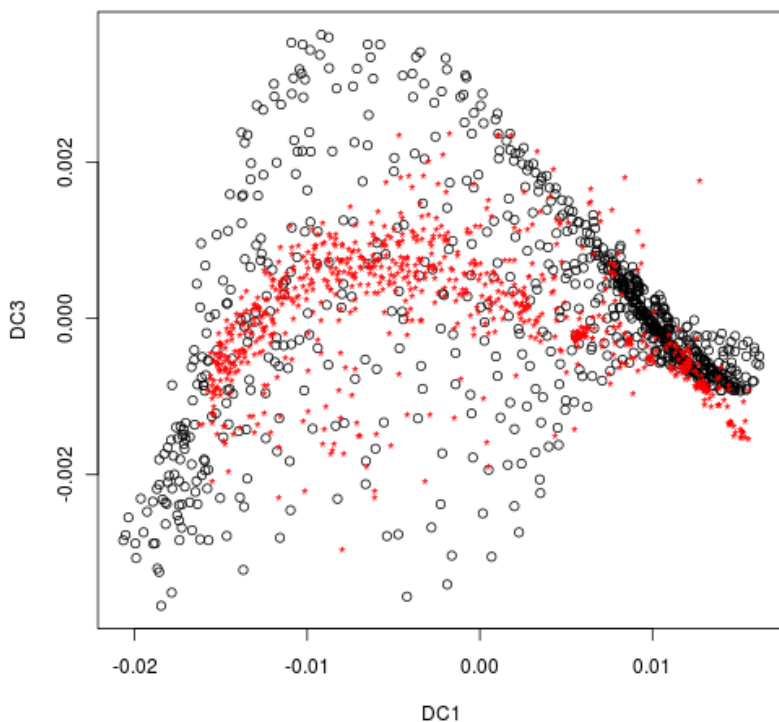


Ilustración 40: Coordenadas de difusión 1 y 3.

Los modelos de Kurucz aparecen en negro, y los de ELODIE en rojo.

En dichas ilustraciones, se muestran los modelos de Kurucz en color negro y las estrellas de ELODIE en rojo. Se puede apreciar que muchas estrellas ELODIE no ocupan exactamente el mismo espacio que los modelos de Kurucz, lo cual no es una buena señal. El motivo no se descubrió, y sea, posiblemente, un problema de preparación de datos no resuelto.

Después, se entrenó un modelo SVM, con las coordenadas de difusión del conjunto completo de modelos de Kurucz, y se validó el modelo con las coordenadas de difusión de las estrellas de ELODIE, ajustando el valor de σ en el kernel de Laplace (valores entre $1e-6$ y $1e-3$), y el valor de ϵ en la construcción del grafo ponderado (entre 2^{-7} y 2^{-3}). Al igual que ocurriera con los experimentos con PCA utilizando los conjuntos completos de Kurucz, los resultados fueron bastante pobres. La Tabla 13 muestra los resultados.

		σ (kernel Laplace)	ϵ (grafo)	Corr	MAE	RMSE	μ_e	σ_e
Rango azul	Teff	0,03125	1e-4	0.6111	3455	4661	2309	4051
	log(g)	0,01562	1e-3	0,6444	0,7422	0,8669	-0,544	0,6748
Rango rojo	Teff	0,01562	1e-4	0,5231	1648	4187	195	4185
	log(g)	0,00781	1e-3	0,447	0,7987	0,9512	-0,483	0,819

Tabla 13: Resultados sobre conjunto de test ELODIE, usando *Diffusion Maps* y SVM. La validación se hizo sobre el propio conjunto de ELODIE. Se utilizó el conjunto completo de Kurucz y una muestra aleatoria de ELODIE.

Diffusion coordinates, Teff < 10000K, Kurucz (black) and ELODIE (red)

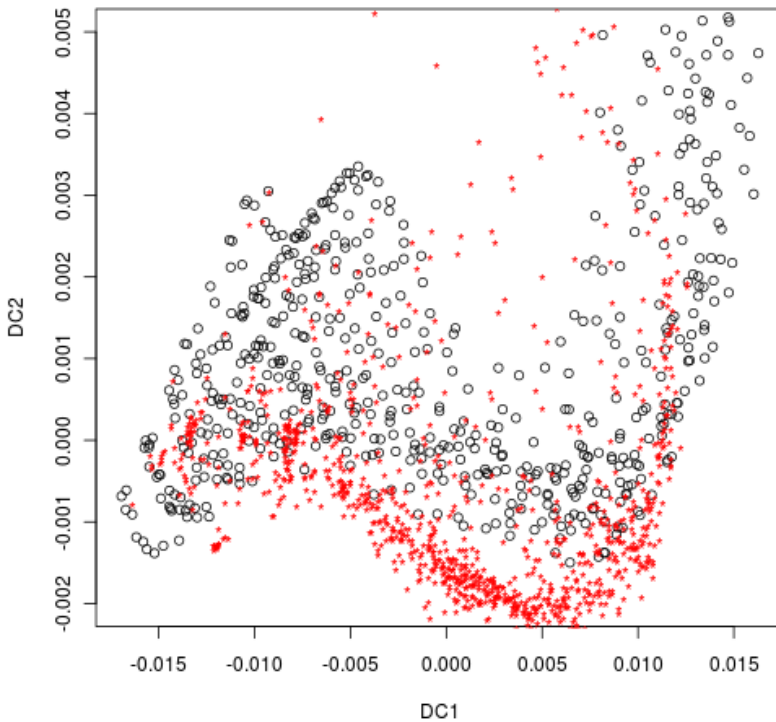


Ilustración 41: Coordenadas de difusión 1 y 2 para estrellas menores de 10000K.

Las estrellas de ELODIE (en rojo) aparecen fuera del espacio que ocupan los modelos de Kurucz (en negro).

Debido a los malos resultados con los conjuntos completos de estrellas, se utilizó la misma estrategia: eliminar de la selección de estrellas aquéllas cuya temperatura fuera mayor de 10013 K. Se formó un conjunto de 1660 estrellas, utilizado 593 estrellas de los modelos de Kurucz, y 1067 estrellas del conjunto ELODIE. Las coordenadas de difusión que se generaron para los dos conjuntos también sufrieron el mismo problema: algunas coordenadas de ELODIE parecían ocupar un espacio diferente que el de Kurucz, véase la Ilustración 41. Al igual que en el caso anterior, se validó el modelo con las coordenadas de difusión de las estrellas de ELODIE, buscando el mejor valor de σ en el kernel de Laplace (valores entre $1e-6$ y $1e-3$), y el mejor valor de ϵ en la construcción del grafo ponderado (entre 2^{-6} y 2^{-3}).

Sin embargo, en este último caso, se consiguió obtener resultados mejores sólo para el parámetro de la temperatura efectiva en el rango azul de longitudes de onda, mientras que el resto de parámetros ofrecieron resultados todavía peores, véase la Tabla 14. Con todo, si se comparan la predicción de T_{eff} en el rango azul con los resultados de la Tabla 10, todavía existe una gran diferencia (RMSE de de 910 K frente a los 352 K de PCA y SVM)

		σ (kernel Laplace)	ϵ (grafo)	Corr	MAE	RMSE	μ_e	σ_e
Rango azul	Teff	0,03125	1e-5	0,7601	739	910	446	793
	log(g)	0,015625	1e-5	0,5738	1,14	1,27	-1,04	0,727
Rango rojo	Teff	0,015625	1e-5	0,3365	1664	5116	-1118	4994
	log(g)	0,125	1e-3	0,5462	0,7887	1	-0,541	0,842

Tabla 14: Resultados sobre conjunto de test ELODIE, utilizando *Diffusion Maps* y SVM. La validación se hizo sobre el propio conjunto de ELODIE. Tanto en el conjunto de Kurucz como en el de ELODIE, Se utilizaron estrellas cuya temperatura efectiva era menor de 10013 K.

Ya se ha visto que existen coordenadas que se salen del espacio de Kurucz (DC2) pero otras parecen estar dentro (DC1). Así que se optó por hacer, al igual que con PCA, una selección de coordenadas de difusión que pudiera mejorar el resultado. Para ello se exportaron las coordenadas de difusión a Weka, y se utilizó el algoritmo de búsqueda de mejores atributos *ClassifierSubsetEval*, una versión simplificada del algoritmo *WrapperSubsetEval*, utilizado en el apartado 3.3.1.4. Este algoritmo es mucho más rápido, a costa de no utilizar la validación cruzada para la evaluación de cada subconjunto de atributos. El sub-algoritmo de evaluación utilizado por *ClassifierSubsetEval* fue *SMOReg* (la implementación en Weka de la regresión SVM), y el sub-algoritmo de selección de atributos *BestFirst*. La validación de las mejores coordenadas de difusión se hizo con el conjunto de ELODIE. Las coordenadas seleccionadas fueron:

Rango azul	Teff	1,3,7,9,10,13,21,23,28
	log(g)	3,7,9
Rango rojo	Teff	1,2,3,4,5,6,7,8,9,10,11,12,26
	log(g)	9,10,11,18,26

Al igual que ocurriera en los experimentos con PCA, los errores disminuyeron notablemente para T_{eff} en el rango azul, si bien dicha mejora no llegó a alcanzar el nivel de otras técnicas. Véase la Tabla 15: el RMSE fue de 658 K, frente al RMSE de 352 K obtenido utilizando PCA y SVM (Tabla 10), o los 506 K de la técnica de los K-vecinos más próximos (Tabla 12). El parámetro $\log(g)$ también llegó a nivel aceptable, un RMSE de 0,801, que mejora ligeramente los resultados obtenidos en PCA y SVM (RMSE=0,8175).

		σ (kernel Laplace)	ϵ (grafo)	Corr	MAE	RMSE	μ_e	σ_e
Rango azul	Teff	0,01562	1e-5	0,7853	458	658	46,50	657
	$\log(g)$	0,0625	1e-5	0,486	0,57	0,801	-0,164	0,784
Rango rojo	Teff	0,0625	1e-4	0,3523	1644	5248	-1239	5102
	$\log(g)$	0,15625	1e-4	0,142	0,9596	1,105	-0,688	0,864

Tabla 15: Resultados sobre conjunto de test ELODIE, utilizando *Diffusion Maps* y SVM. La validación se hizo sobre el propio conjunto de ELODIE. Tanto en el conjunto de Kurucz como en el de ELODIE, Se utilizaron estrellas cuya temperatura efectiva era menor de 10013 K. Se utilizaron una selección de las 30 primeras componentes principales.

Para los parámetros en el rango rojo, la selección de coordenadas de difusión no consiguió mejorar los resultados.

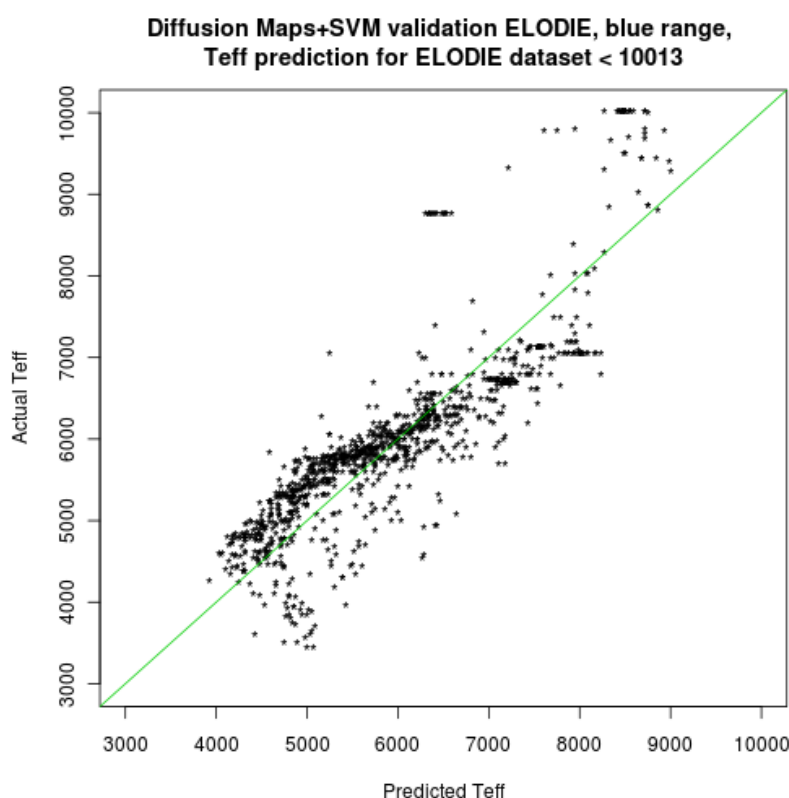


Ilustración 42: Temperatura real vs. predicha utilizando *Diffusion Maps* y SVM, en el rango azul para estrellas < 10013 K.

La validación se realizó con el conjunto ELODIE.

Las ilustración 42 muestra la temperatura efectiva real frente a la predicha. El pequeño grupo de estrellas cuya temperatura es menor de 4000 K se clasifica como estrellas cuya temperatura está en el entorno de los 5000 K, lo cual es hasta cierto punto lógico porque los modelos de Kurucz no tienen temperaturas por debajo de los 4000 K. Hasta los 7000 K las predicciones se comportan razonablemente bien, y a partir de esta temperatura, las predicciones van empeorando progresivamente. Por encima de los 8000 K, muchas predicciones cometen errores de más de 1000 K. Nótese la gran similitud que guarda este gráfico con el obtenido en PCA y SVM para el mismo parámetro y rango de longitudes de onda; véase la Ilustración 27.

La ilustración 43 muestra el logaritmo de la gravedad real frente al predicho. A pesar de tener un RMSE (0,801) cercano a los conseguidos con PCA y SVM(0,8175, véase Tabla 10) , el índice de correlación de Diffusion Maps (0,486) es mucho menor que el obtenido en PCA y SVM (0,8113). Esto es debido a que el modelo regresivo apenas predice estrellas por debajo del valor 3, con lo que la correlación por debajo de esta cifra es muy baja y reduce índice de correlación del conjunto completo. El modelo regresivo es razonablemente bueno para estrellas en el entorno del 4, y mucho peor para estrellas de menos de 3.

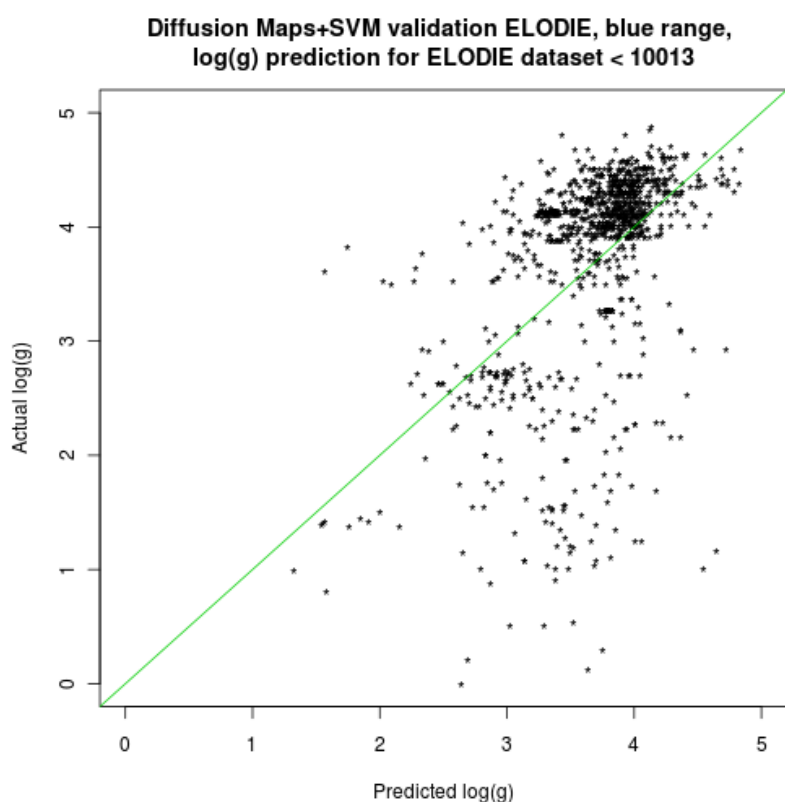


Ilustración 43: $\log(g)$ real vs. predicha utilizando Diffusion Maps y SVM, en el rango azul, para estrellas menores de 10013 K.

La validación se realizó con el conjunto ELODIE.

4.4 Regresión Partial Least Squares

La regresión *Partial Least Squares* (PLSR) (20) es un método para construir modelos regresivos cuando las variables de entrada son muchas y altamente correladas. El objetivo de este método es encontrar componentes que contengan la mayoría de la información de los atributos de entrada (\mathbf{X}) para la predicción de los atributos de salida (\mathbf{Y}). En concreto, descompone \mathbf{X} e \mathbf{Y} con la restricción de que los componentes expliquen la covarianza entre \mathbf{X} e \mathbf{Y} tanto como sea posible. \mathbf{X} se descompone en la forma $(\mathbf{X} = \mathbf{TP}^T)$, donde \mathbf{T} es la matriz *scores* y \mathbf{P} la matriz *loadings*.

Después, la descomposición de \mathbf{X} es utilizada para hacer una regresión para predecir \mathbf{Y} de la siguiente forma

$$\hat{\mathbf{Y}}_k = \beta_{k0} + \beta_{k1} T_1 + \dots + \beta_{kp} T_p \quad \text{donde}$$

$\hat{\mathbf{Y}}_k$ es la estimación de la k -ésima columna \mathbf{Y} para el nuevo espectro observado.

T_i es una combinación lineal de \mathbf{X} , correspondiente a la columna i -ésima de \mathbf{T} .

β_{ki} son los coeficientes de regresión, para la variable de salida Y_k

p es el número de componentes escogidos para el modelo de regresión.

Los coeficientes β_{ki} son estimados por regresión simple tomando como variable independiente T_i y como variable dependiente \mathbf{Y} . La matriz \mathbf{T} contiene muchas menos columnas y concentra la covarianza que hay entre \mathbf{X} e \mathbf{Y} . El número de componentes p (es decir, el número de columnas de \mathbf{T}) debe ser determinado mediante algún método de validación, generalmente, por validación cruzada.

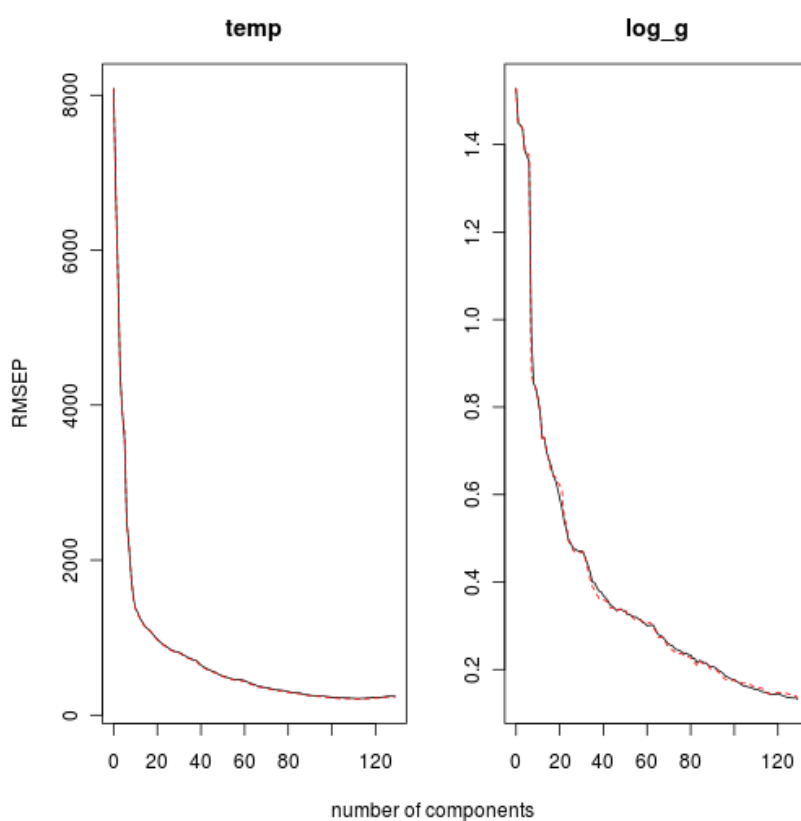


Ilustración 44: RMSE en función del número de componentes en PLS para Kurucz en el rango azul

Después de obtener el modelo predictivo para el conjunto de datos de Kurucz en el rango azul, con un total de 100 componentes, se obtenía una estimación del error cuadrático medio (RMSE), utilizando validación cruzada, de 217 K en Teff y de 0.1666 en log(g). La Ilustración 44 muestra la evolución de RMSE en función del número de componentes para las dos variables de salida estudiadas. La Ilustración 45 muestra esta misma información para el rango rojo.

En el rango azul, se observa que el error desciende rápidamente en los primeros primeros componentes, y llega a estabilizarse en log(g) a partir de los 129 componentes (mínimo RMSE=0.1284) y en Teff a partir de los 100 componentes (mínimo RMSE=238.7). En el rango

rojo son similares; 142 componentes para $\log(g)$ (mínimo RMSE=0.1601) y 87 componentes para T_{eff} (mínimo RMSE de 502 K).

Al tratar de aplicar este modelo predictivo a los espectros reales de ELODIE, el número de componentes óptimo difirió respecto al conjunto de datos de Kurucz. La Ilustración 46 muestra la relación entre el RMSE y el número de componentes para el conjunto de datos de ELODIE, en el rango azul. Se observa que el número de componentes óptimo es mucho menor en ambos parámetros (4 en T_{eff} , y 22 para $\log(g)$). La Tabla 16 muestra los índices de error para el conjunto ELODIE, en el rango azul, utilizando el número óptimo de componentes para ELODIE. Nótese que el error cometido es mucho mayor para T_{eff} (RMSE=3322 K), mientras que para $\log(g)$ alcanza un valor aceptable (RMSE=0.6629), mejor que lo obtenido con PCA y SVM (véase Tabla 10). La Ilustración 47 muestra la relación entre $\log(g)$ predicho y real, y se puede apreciar que el modelo regresivo no es tan bueno para valores por debajo de 2. En el rango rojo, los errores son mucho peores que el azul y por ello no se muestran, aunque la relación RMSE y número de componentes es similar a la del rango azul.

		Núm. comps.	Corr	MAE	RMSE	μ_e	σ_e
Rango azul	Teff	4	0,8124	2015	3322	-738	3239
	log(g)	22	0,775	0,5054	0,6628	-0,025	0,6626

Tabla 16: Resultados de aplicar regresión PLS en el rango azul sobre el conjunto de ELODIE. La validación se hizo sobre ELODIE.

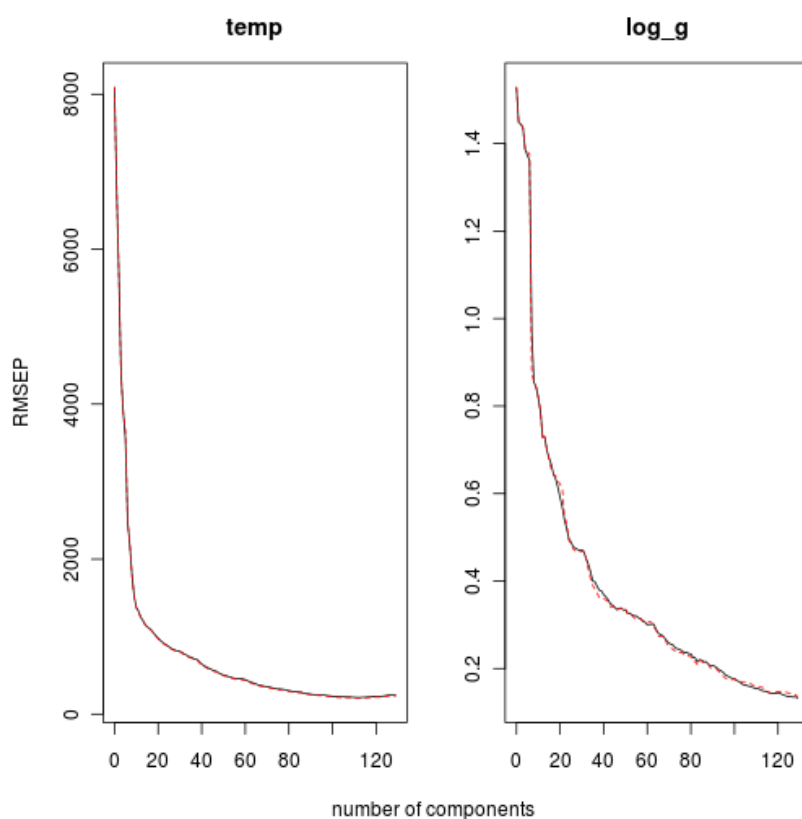


Ilustración 45: RMSE en función del número de componentes en PLS para Kurucz en el rango rojo

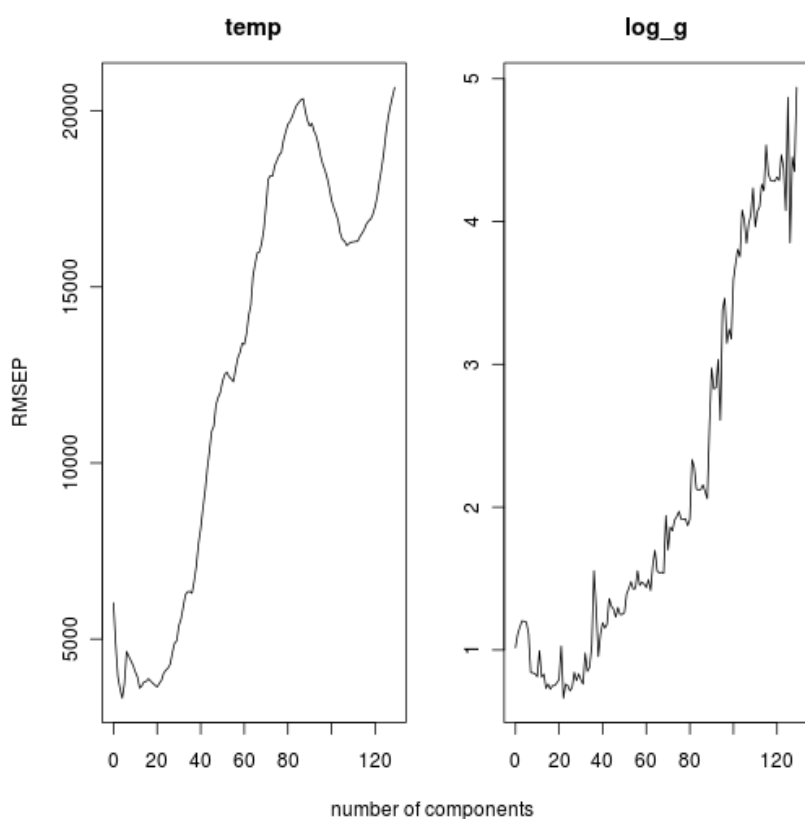


Ilustración 46: RMSE vs. número de componentes en PLS para ELODIE, rango azul.

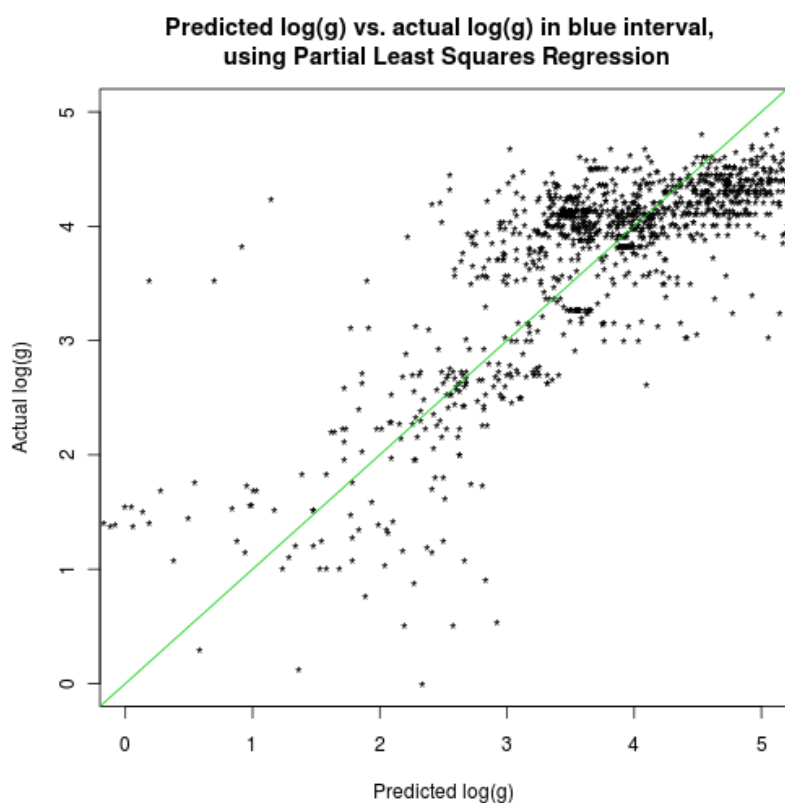


Ilustración 47: $\log(g)$ predicho vs. real en el rango azul, utilizando PLSR .

Se utilizaron 22 componentes, el número óptimo para el conjunto ELODIE.

En conclusión, la regresión PLS no ha resultado un buen método para T_{eff} , aunque ha conseguido resultados moderadamente aceptables para $\log(g)$ en el rango azul. PLS sobreajusta los datos para Kurucz, y prueba de ello es la gran diferencia entre el número de componentes óptimos con validación cruzada con Kurucz y cuando se valida con ELODIE. Por otra parte, en (18) se advierte de que este algoritmo debe ser utilizado cuando se estime que existe una dependencia lineal o cuasi-lineal entre los datos de entrada y los de salida. Si la dependencia entre ambos conjuntos no es lineal, la regresión PLS tiende a sobreajustar los datos. Esto es lo que parece que ocurre en el experimento realizado, lo cual indica que la dependencia de la temperatura efectiva no sea lineal. El mismo artículo propone una variante de PLS, a la que denomina *Kernel PLS*, que transforma el espacio de entrada en un espacio de características (a través de una función núcleo o *kernel*), en el que la dependencia entre los conjuntos de entrada y de salida podría ser lineal. El uso de funciones núcleo se utilizó en los experimentos de PCA y SVM. Desgraciadamente, no se pudieron hacer experimentos con *Kernel PLS* al no encontrar implementaciones de este algoritmo en R.

4.5 Entrenamiento con ELODIE

El objetivo principal del trabajo era la estimación de temperatura y gravedad de espectros reales a partir de los espectros sintéticos de Kurucz. PCA y SVM fue la técnica con la que mejores resultados se habían obtenido. Con propósitos comparativos, se volvió a utilizar esta técnica utilizando como conjunto de entrenamiento el conjunto de estrellas de ELODIE para generar un modelo predictivo para FLAMES. Para ello, se utilizaron las 30 primeras componentes principales de ELODIE para entrenar una máquina de vectores soporte (SVM) que se utilizó para predecir T_{eff} y $\log(g)$ en los conjuntos de datos FLAMES I y II.

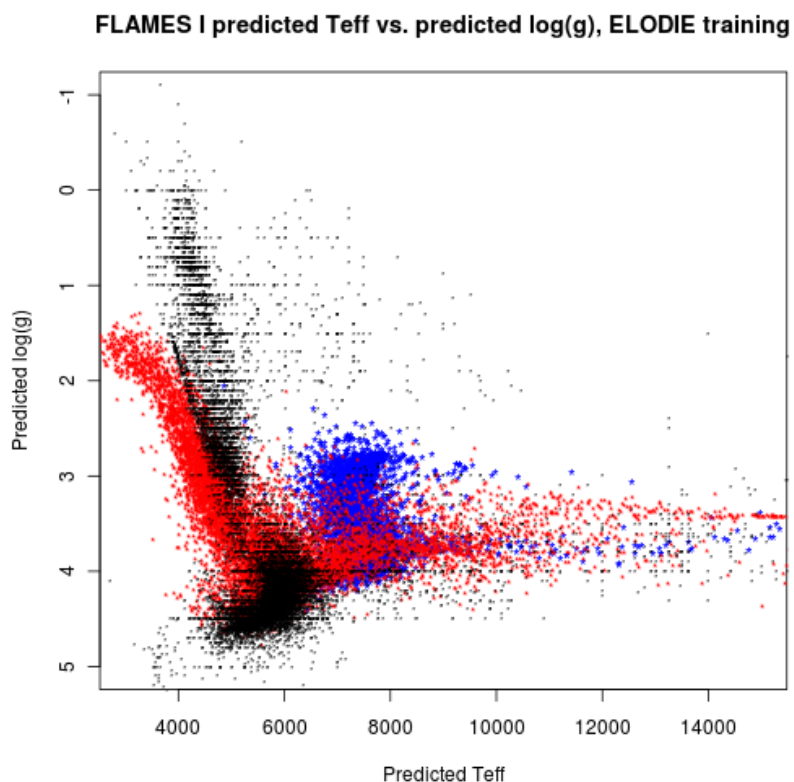


Ilustración 48: Predicción de T_{eff} y $\log(g)$ en FLAMES I, entrenando con ELODIE

La Ilustración 48 muestra T_{eff} frente a $\log(g)$ para el conjunto de datos FLAMES I, en azul las predicciones hechas en el rango de longitudes de onda azul, y en rojo las del rango rojo. Se

incluyen, en color negro, las temperaturas y gravedades del conjunto B/PASTEL (10), conjunto de 15.495 estrellas reales cuyas temperaturas y gravedades han sido recopiladas de diversas fuentes y cuya veracidad se da por cierta. La Ilustración 49 muestra esta misma información para las predicciones en el conjunto de datos FLAMES II.

Tanto en FLAMES I como en FLAMES II, el rango azul no parece predecir correctamente ninguno de los parámetros, a tenor de la distribución de los valores de dichos parámetros. En el rango rojo, parece que en el entorno de los 6000K de Teff y 4 dex de log(g) la distribución de Teff y log(g) coincide. Sin embargo, entre los 4000K y 5000K, las predicciones de Teff parecen sufrir un sesgo negativo.

Es curioso que sea el rango rojo el que mejor predice cuando se entrena con ELODIE, siendo el peor cuando se entrena con Kurucz. En cualquier caso, con este experimento no se trató de hacer un estudio en profundidad, su propósito fue más un acercamiento al entrenamiento con ELODIE sin entrar en detalles. Al contrario que con el entrenamiento con modelos de Kurucz, no se restringió el conjunto de datos a las estrellas *frías*, ni se realizó una selección de mejores componentes principales.

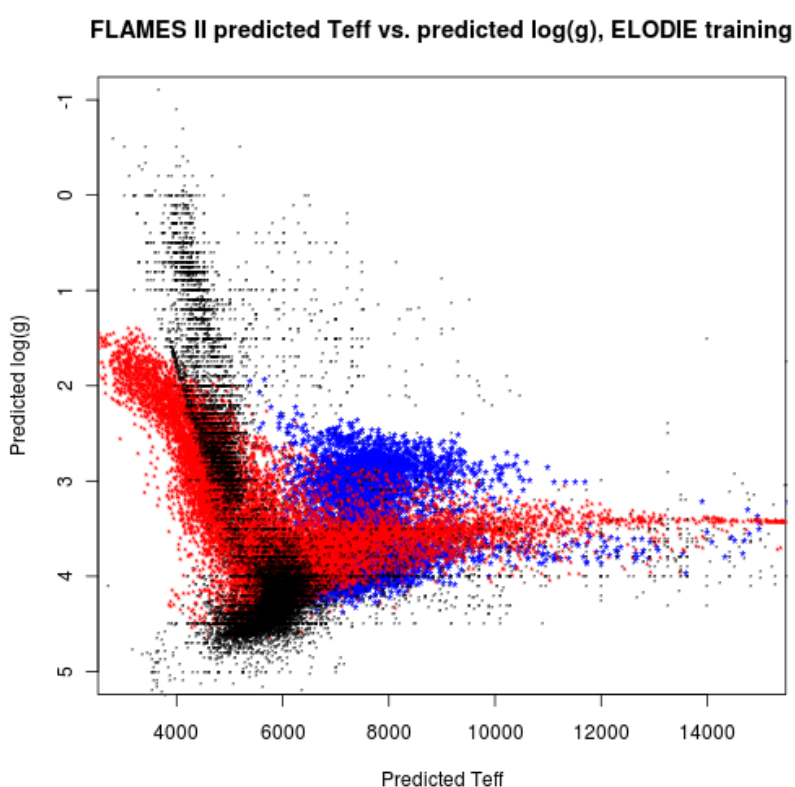


Ilustración 49: Predicción de Teff y log(g) en FLAMES II, entrenando con ELODIE

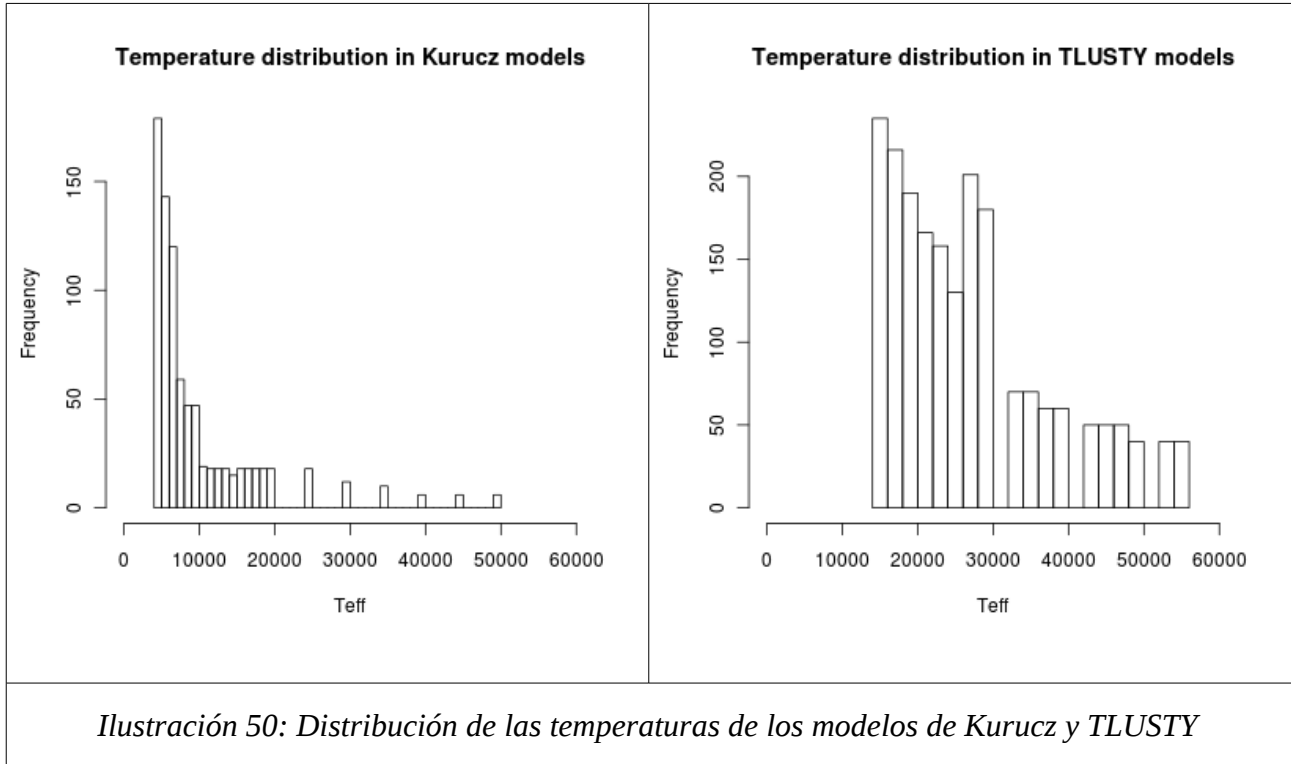
4.6 Entrenamiento con TLUSTY

Los modelos de Kurucz no resultaron un buen conjunto de entrenamiento para estrellas superiores a 10000K. Se conocía de antemano que estos modelos no eran adecuados para estrellas con temperaturas altas, y también se observó que en dicho conjunto de datos los modelos de estrellas calientes eran mucho más escasos que los de estrellas más frías.

Con objeto de experimentar con otros modelos de estrellas, se escogió TLUSTY (9), un conjunto de modelos específico para estrellas calientes. El rango de temperaturas de dicha familia de modelos va desde los 15000K hasta los 55000 K, y contiene una mayor densidad de modelos de estrellas en

temperaturas altas (Ilustración 50).

Las primeras pruebas se realizaron utilizando PCA y SVM con las 30 primeras componentes principales de los modelos TLUSTY en los dos rangos de longitud de onda. La validación de los modelos regresivos se realizó mediante validación cruzada de 10 subconjuntos. Los modelos generados dieron muy buenos resultados con el conjunto de entrenamiento, véase la Tabla 17.



		σ (kernel Laplace)	Corr	MAE	RMSE	μ_e	σ_e
Rango azul	Teff	0,0625	0,9999	17	107	-10	106
	log(g)	0,0625	0,9999	0,00029	0,0024	-0,00001	0,0024
Rango rojo	Teff	0,0625	0,9998	16,78	195	-13	195
	log(g)	0,0625	0,9999	0,0009	0,0095	-0,0006	0,009

Tabla 17: Resultados sobre TLUSTY, utilizando las 30 primeras componentes principales y SVM, con validación cruzada de 10 subconjuntos sobre TLUSTY.

Sin embargo, al utilizar estos mismos modelos con el conjunto de estrellas de ELODIE cuya temperatura efectiva era mayor de 15000 K, los resultados no fueron buenos, véase Tabla 18.

Es importante tener en cuenta que, a pesar de que el RMSE de log(g) en el rango azul es bajo con respecto a otros experimentos, el índice de correlación cercano a cero indica que la predicción no es tan buena. Esto se debe a que el conjunto de estrellas de ELODIE evaluado se compone de estrellas cuyos valores de log(g) oscilan entre 2,6 y 4,5, pero que concentran la mayor parte de los valores entre 3.0 y 4.0., mientras que el conjunto de entrenamiento (TLUSTY) contiene modelos entre 1.5 y

4.75. Sería necesario que el conjunto de test tuviera estrellas cuyos valores de $\log(g)$ se distribuyeran de forma más parecida a la de TLUSTY, para observar si el modelo regresivo tiene la misma precisión en valores menores de 3.0 y mayores que 4.0. Por tanto, no se pueden considerar como representativos los valores de error para $\log(g)$ en la Tabla 18.

		σ (kernel Laplace)	Corr	MAE	RMSE	μ_e	σ_e
Rango azul	Teff	0,0625	0,739	6174	7408	1160	7384
	$\log(g)$	0,0625	0,03	0,4311	0,4815	-0,139	0,465
Rango rojo	Teff	0,0625	0,132	8022	9427	1984	9301
	$\log(g)$	0,0625	0,173	0,474	0,619	-0,322	0,534

Tabla 18: Resultados sobre ELODIE, utilizando las 30 primeras componentes principales y SVM, con validación cruzada sobre el conjunto TLUSTY.

La gran diferencia entre los valores predichos en el conjunto de entrenamiento y el de test demuestra que los modelos regresivos están claramente sobreajustados, a pesar de que hayan sido obtenidos mediante validación cruzada.

En un segundo intento para conseguir buenos resultados, se utilizaron las mismas técnicas de vecino más próximo expuestas en el punto 4.2 para predecir los parámetros físicos de las estrellas de ELODIE con Teff mayor de 15000K. En todas las predicciones los errores fueron muy altos, con índices de correlación cercanos al cero, RMSE superiores a 10000 K para Teff y cercanos a 1 o superiores para $\log(g)$.

5 Comparación de resultados

Los artículos expuestos en el apartado 2 tienen el mismo objetivo planteado en el presente trabajo de fin de máster (TFM), pero los medios que se utilizan para llevarlo a cabo dan como resultado unos errores que podrían no ser comparables con los obtenidos en este trabajo.

En (1) se utilizan el mismo conjunto de espectros reales para hacer la predicción y la evaluación, un conjunto seleccionado con alto ratio de señal/ruido, y que no contiene estrellas binarias. Además, en sus predicciones se utiliza un solo rango de longitudes de onda que cubre los dos rangos que se han utilizado en este trabajo. La estrategia que se sigue es la del vecino más cercano, y por lo tanto no se genera un modelo predictivo como tal, sino que se utilizan las estrellas más cercanas a una dada para calcular los parámetros físicos. La evaluación se realiza extrayendo una estrella de dicho conjunto y prediciendo sus parámetros físicos comparando su espectro con el resto de espectros del conjunto, un método conocido como *leave-one-out*. La desviación estándar del error es de 86 K, 0.28 dex para Teff y $\log(g)$, respectivamente.

En (2) se utiliza el mismo conjunto como entrenamiento MARCS, un conjunto de espectros sintéticos. Como conjunto de validación, se utiliza el propio conjunto MARCS, con temperaturas entre 4000 K y 8000 K, y aplicando diversos niveles de ruido a los espectros. Solo se proporcionan dos ejemplos de estrellas reales. El error para los espectros sintéticos fue de entre 50K y 150K para Teff, y entre 0.04 y 0.2 para $\log(g)$, para estrellas con un SNR de 50.

En (5), el conjunto de datos de entrenamiento es MILES, un conjunto de espectros reales. En este

caso sí se utiliza un conjunto de estrellas reales ajenas al conjunto de entrenamiento, una selección del conjunto ELODIE con índice de calidad mayor o igual a 1. Los resultados fueron de $\mu = -3.68\%$ y $\sigma = 5.20\%$ para T_{eff} , y de $\mu = 0.1167$ dex y $\sigma = 0.4864$ dex para $\log(g)$

Para hacer la comparación, se ha optado por distinguir entre los métodos predictivos cuyas validaciones utilizan el mismo conjunto de entrenamiento (Tabla 19) y las que utilizan conjuntos de datos diferentes para el entrenamiento y la validación (Tabla 20).

Método de predicción	Conjunto entrenamiento	Conjunto de validación	Método de evaluación	Teff		log(g)	
				Sesgo	Desviación sesgo	Sesgo	Desviación sesgo
Artículo (1)	ELODIE	ELODIE	<i>Leave one out</i>	17	86	0.04	0,28
MATISSE (2)	MARCS	MARCS	Ruido gaussiano	50K-130K (sesgo +suma)		0,06-0,16 (sesgo +suma)	
TFM azul PCA+SVM	Kurucz <10000K	Kurucz <10000K	Validación cruzada	-15	98	-0,0059	0,1108
TFM rojo PCA+SVM	Kurucz <10000K	Kurucz <10000K	Validación cruzada	-25	168	0,0135	0,1612

Tabla 19: Comparación entre métodos predictivos con el mismo conjunto de validación que de entrenamiento

Método de predicción	Conjunto entrenamiento	Conjunto de validación	Método de evaluación	Teff		log(g)	
				Sesgo	Desviación sesgo	Sesgo	Desviación sesgo
<i>Piecewise PLS</i> (5)	MILES	ELODIE <15000K	Conjuntos diferentes	-378	520	0,1167	0,4864
TFM azul PCA+SVM	Kurucz <10000K	ELODIE <10000K	Conjuntos diferentes	-30	351	-0,5469	0,6079
TFM rojo PCA+SVM	Kurucz <10000K	ELODIE <10000K	Conjuntos diferentes	-155	488	-0,373	0,7139
TFM azul PLS	Kurucz	ELODIE	Conjuntos diferentes	-738	3239	-0,025	0,6626

Tabla 20: Comparación entre métodos predictivos cuyo conjunto de validación difiere del de entrenamiento

En la Tabla 19 se comparan los métodos que utilizan el mismo conjunto de datos para entrenamiento y para validación. Para Teff, el método de validación cruzada con Kurucz y ELODIE dan un resultado muy parecido. El menor error para el $\log(g)$ se produce con la validación cruzada en Kurucz, aunque MATISSE puede ser mejor o peor dependiendo del tipo de estrellas

evaluado para cualquiera de los parámetros físicos. No se puede ser categórico sobre cuál es el mejor método por la escasa diferencia entre los errores, y porque los conjuntos de datos son diferentes en los distintos métodos. Además, los modelos se han validado de formas diferentes. En (19) se critica el método *leave-one-out* de (1) porque las predicciones pueden tener una gran varianza, y se recomienda para la validación de conjuntos más pequeños de datos. Por otra parte, la validación en MATTISE consiste en la degradación del propio conjunto de entrenamiento con ruido gaussiano, un método que puede resultar demasiado optimista.

Si se utilizan conjuntos de procedencias diferentes para el entrenamiento y la validación el resultado debería ser más confiable. En la Tabla 20, para el parámetro T_{eff} , el menor error se produce para el presente TFM en el rango azul de longitudes de onda, si bien el conjunto de validación es más restringido que para PLS. Los experimentos del TFM con PCA+SVM y PLSR presentan mayores desviaciones respecto al sesgo para $\log(g)$ en cualquiera de los dos rangos de longitudes de onda respecto a *Piecewise PLS*. En cambio, en el caso de los experimentos de este TFM con PLSR, el sesgo es menor, aunque la desviación respecto al sesgo no lo sea. Sería necesario utilizar otros índices de error u otros tipos de pruebas para conocer el mejor de ambos métodos.

6 Conclusiones y mejoras

Este TFM ha supuesto el primer trabajo de investigación del alumno. Durante los dos años que ha durado su desarrollo se han realizado muchos experimentos, y su resultado ha sido plasmado en este documento de forma resumida, pero sin dejar de exponer todo lo realizado. Se pueden destacar dos fases en su desarrollo: la preparación de los datos y la minería de datos.

La fase de preparación de los datos ha supuesto un esfuerzo equiparable al de minería de datos, a pesar de que aparentemente pueda parecer más sencillo. Sobre esta fase, se pueden enumerar, esquemáticamente, las siguientes conclusiones:

- Si no se seleccionan, limpian y armonizan los datos antes de la minería de datos, no se pueden obtener buenos resultados. En el caso de este trabajo, fue necesario el consejo del tutor del TFM para conocer qué aspectos de los datos espectrales había que preparar: la interpolación, la sustracción del continuo, la normalización, eliminación de ruido.
- Uno de los aspectos que no fueron contemplados al principio, el efecto Doppler, resultó ser muy importante para conseguir un conjunto homogéneo de espectros reales. Este problema se descubrió al comparar las componentes principales de los espectros reales con los espectros sintéticos de Kurucz, y comprobar que no ocupaban el mismo espacio. Después de estudiar los espectros que se salían de este espacio, se descubrió que la mayoría de ellos presentaban un desplazamiento hacia un lado o a otro, consecuencia del efecto Doppler.
- Los rayos cósmicos necesitan un tratamiento especial, ya que no son suprimidos con las técnicas habituales de supresión de ruido.

Es en la minería de datos donde se evalúan los modelos predictivos, y donde se pueden comparar entre ellos. Las conclusiones en esta fase fueron las siguientes:

- Cuando se entrenan modelos con el conjunto de de datos de Kurucz, los modelos generados producen un menor error en el rango azul de longitudes de onda.
- En los modelos entrenados con los modelos de Kurucz, restringir el conjunto de estrellas a aquéllas cuya temperatura está por debajo de los 10.000 K hace que el modelo de predicción de la temperatura efectiva mejore notablemente en ambos rangos de longitud de onda. Esto

ocurre en PCA+SVM, en *diffusion maps* y en las técnicas de vecindad. El motivo se debe a que, en Kurucz, los modelos de estrellas calientes eran mucho más escasos que los de estrellas más frías. Este fue el motivo para experimentar con TLUSTY, un conjunto de espectros con mayor abundancia de estrellas de temperaturas mayores de 15000 K.

- Se pueden mejorar los modelos predictivos si, en vez de seleccionar los n primeros componentes principales en PCA (o coordenadas de difusión en *diffusion maps*), se realiza una búsqueda de los mejores componentes principales (o mejores coordenadas de difusión) aunque estos componentes no sean consecutivos. En PCA, los errores del logaritmo de la gravedad en ambos rangos y la temperatura efectiva en el rango rojo mejoraron notablemente, y en *diffusion maps*, en temperatura y gravedad solamente en el rango azul.
- En PCA, la validación de los modelos predictivos con el conjunto de ELODIE da como resultado que cambie el valor de σ del kernel de Laplace a un valor menor y mejora ligeramente los resultados en ELODIE.
- La regresión PLS ofrece resultados moderadamente aceptables para el logaritmo de la gravedad en el rango azul.

Hay, sin embargo, muchos trabajos en los que no se han obtenido buenos resultados, posiblemente porque no ha habido tiempo para explorarlos con la suficiente profundidad. En *diffusion maps* se descubrió que algunas coordenadas de difusión del conjunto ELODIE no ocupaban el mismo lugar que las de Kurucz, quizá debido a algún problema de preparación de datos aun no descubierto. Es seguro que la adecuación de espacios de ambos conjuntos sea un buen indicador de la calidad de la fase del preprocesado, por lo que en futuros trabajos sería muy conveniente idear un índice que exprese el grado en el que los datos de test (ELODIE) se encuentran dentro de los de entrenamiento (Kurucz).

En el caso de los K-vecinos más próximos, los espectros se comparan directamente. Para buscar la causa de las malas predicciones, se podrían haber mostrado los espectros de mayor error junto a: a) los espectros cercanos a partir de los que se ha hecho la predicción, y b) los espectros que se encuentran cerca de sus temperatura real. Estas comparaciones podrían aclarar estas diferencias (quizá con la ayuda de un buen intérprete de espectros estelares), tanto para los datos predichos con errores elevados en Kurucz, como para el entrenamiento con TLUSTY. Por la experiencia adquirida durante este trabajo, fácil sería que dichas diferencias se deban a algún otro aspecto no contemplado durante la preparación de los datos.

Otro aspecto no explorado es la investigación de una medida de distancia distinta las utilizadas que haga que los espectros con parámetros físicos cercanos estén más cerca y alejen los que se encuentran más lejos. Esta medida de distancia podría ser aplicada tanto a técnicas de vecindad como a *diffusion maps*.

Aunque el entrenamiento de modelos con ELODIE no formaba parte de los objetivos del trabajo, se realizaron este tipo de pruebas con propósitos comparativos. A tenor de los resultados obtenidos en la predicción de los conjuntos FLAMES, esta posibilidad podría ser prometedora en futuras investigaciones. Posiblemente, una selección de componentes principales podría mejorar los resultados.

Apéndice I. Fórmulas de medida del error

Sean $P = \{p_1, p_2 \dots p_n\}$ el conjunto de valores predichos para un determinado parámetro en un

número n de instancias. Sean $A = \{a_1, a_2 \dots a_n\}$ el conjunto de valores reales para un dicho parámetro. El error cometido se define como el conjunto $\{(p_1 - a_1), (p_2 - a_2) \dots (p_n - a_n)\}$, conjunto que por abreviar se designará por la letra e : $\{e_1, e_2, \dots e_n\}$

A continuación se definen las siguientes medidas de error utilizadas en el presente documento.

Índice de correlación (Corr). Rango de -1 a 1, con el siguiente significado: 1 – P y A tienen una fuerte correlación positiva. 0 – P y A son independientes -1 – P y A tienen una fuerte correlación negativa.	$Corr = \frac{\sum_{i=1}^n (p_i - \bar{p})(a_i - \bar{a})}{\sqrt{\sum_{i=1}^n (p_i - \bar{p})^2 \sum_{i=1}^n (a_i - \bar{a})^2}}$
Error Medio Absoluto (MAE)	$MAE = \frac{1}{n} \sum_{i=1}^n e_i $
Raíz del error cuadrático medio (RMSE)	$RMSE = \sqrt{\frac{\sum_{i=1}^n e_i^2}{n}}$
Sesgo	$\mu_e = \frac{1}{n} \sum_{i=1}^n e_i$
Desviación típica del error	$\sigma_e = \sqrt{\frac{\sum_{i=1}^n (e_i - \mu_e)^2}{n-1}}$

Bibliografía

- (1) Katz, D.; Soubiran, C.; Cayrel, R.; Adda, M.; Cautain, R.; *On-line determination of stellar atmospheric parameters*; Astronomy and Astrophysics, 338: 151-160; 1998.
- (2) Recio-Blanco, A.; Bijaoui, A.; de Laverny, P.; *Automated derivation of stellar atmospheric parameters and chemical abundances: the MATISSE algorithm*; Monthly Notices of the Royal Astronomical Society, 370: 141–150; 2006.
- (3) <http://www.eso.org/sci/facilities/paranal/instruments/flames>
- (4) Richards, J.W.; Freeman, P.E.; Lee, B.; Schafer, C.M.; *Exploiting Low-Dimensional Structure in Astronomical Spectra*; The Astrophysical Journal, 691: 32-42; 2009.
- (5) Jian-Nan Zhang; A-Li Luo; Yong-Heng Zhao; *Automated estimation of stellar fundamental parameters from low resolution spectra: the PLS method*. Research in Astron. Astrophys, Vol 9, No. 6, 712-724; 2009.
- (6) Sánchez-Blázquez, P; Peletier, R.F.; Jiménez-Vicente, J. et al; *Medium-resolution Isaac Newton Telescope library of empirical spectra* ; Mon. Not. R. Astron. Soc. 371, 703–718. 2006.
- (7) Dayal, B.S.; MacGregor, J.F; *Improved PLS algorithms*; Journal of Chemometrics, Vol 11, 73-85; 1997
- (8) Tennyson, J. *Astronomical Spectroscopy. An introduction to the Atomic and Molecular*

Physics of Astronomical Spectra; Imperial College Press Advanced Physics Texts – Vol. 2; ISBN 1-86094-513-9 ; pp 3-6; 2005

- (9) Hubeny, I.; Lanz, T.; *NLTE line blanketed model atmospheres of hot stars. I. Hybrid Complete Linearization/Accelerated Lambda Iteration Method* ; Astrophysical Journal, pp 439, 875; 1995 (Véase <http://nova.astro.umd.edu/Tlusty2002/tlusty-frames-refs.html>)
- (10) Soubiran, C.; Le Campion, J.-F., Cayrel de Strobel, G.; Caillo, A.; *The PASTEL catalogue of stellar parameters*; Astronomy & Astrophysics manuscript no. pastel v3.; 2010 (Véase <http://vizier.u-strasbg.fr/viz-bin/VizieR?-source=B/pastel>)
- (11) [Moultaka, J.](#); [Ilovaisky, S. A.](#); [Prugniel, P.](#); [Soubiran, C.](#); *The ELODIE Archive*; Publications of the Astronomical Society of the Pacific Vol. 116, No. 821, pp. 693-698 ; 2004 . (véase <http://www.obs-hp.fr/www/guide/elodie/elodie-eng.html>)
- (12) Donoho, D.L.; Johnstone, I.M.; *Ideal spatial adaptation by wavelet shrinkage*; Biometrika , 81. pp 425-55; 1994
- (13) Hernández Orallo , J; Ramírez Quintana, M.J.; Ferri Ramírez, C; *Introducción a la Minería de Datos*; Pearson Educación, S.A.; pp . 353-382; 2004
- (14) Smola, A.J.; Schölkopf, B; *A tutorial on Support Vector Regression*; NeuroCOLT2 Technical Report Series; 1998
- (15) Iba, W; Langley, P.; *Induction of One-Level Decision Trees*; Proceedings of the Ninth International Conference on Machine Learning; 1992
- (16) Hechenbichler, K. ; Schliep, K.; *Weighted k-Nearest-Neighbor Techniques and Ordinal Classification*. Collaborative Research Center 386, Discussion Paper 399; 2004
- (17) Kohavi, R; John, G. H.; *Wrappers for feature subset selection*. Artificial Intelligence, 97(1-2): pp 273-324. ; 1997
- (18) Rosipal, R; Trejo, L.J.; *Kernel Partial Least Squares Regression in Reproducing Kernel Hilbert Space* ; Journal of Machine Learning Research 2 97-123 ; 2001)
- (19) Refaeilzadeh, P; Tang, L; Liu., H.; *Cross Validation* . Encyclopedia of Database Systems, Editors: M. Tamer Ozsu and Ling Liu. Springer; 2009.
- (20) Garthwaite, P.H.; *An Interpretation of Partial Least Squares*; Journal of the American Statistical Association, Vol. 89, no. 425, Theory And Methods; 1994