Comparación de métodos de clasificación de curvas de luz de estrellas periódicas

Ignacio Haeussler

29 de noviembre de 2017

1. Introducción

1.1. Motivación

El estudio de las estrellas variables (estrellas que varían su brillo con respecto a la tierra) está en el núcleo de muchas investigaciones centrales en astrofísica: las fuentes pulsantes ponen a prueba las teorías de estructura y evolución estelar, los sistemas eruptivos y episódicos nos informan sobre el nacimiento, crecimiento y pérdida de masa de las estrellas, y los sistemas eclipsantes restringen su transferencia de masa, la evolución binaria, demografía exoplanetaria y relación masa, radio y temperatura. Algunos de los sistemas eclipsantes y muchos de los más comunes sistemas pulsantes son los medios fundamentales para determinar distancias precisas a clusters, flujos de satélites perturbados alrededor de la vía láctea y al grupo local de galaxias.

Por otro lado, la clasificación de estrellas variables es un paso importante para dejarlas disponibles para investigación futura: la selección en grandes datasets es mucho más eficiente si existe una clasificación previa. Además, el definir clases científicas en términos de parámetros físicos entrega una visión estadísticamente no sesgada sobre los mecanismos de variación y sobre los bordes de bandas de inestabilidad presentes en el espacio de las variables.

1.2. Objetivo

Este informe presenta los resultados obtenidos de la utilización de dos métodos de clasificación sobre las clases de estrellas variables RR Lyrae, Cefeidas y binarias eclipsantes: MLP y Random

Forest. La comparación se realizó en base a la exactitud (accuracy)y grado de overfitting. Además, análisis varios son realizados sobre las características utilizadas para la clasificación de las curvas de luz, y sobre heurísticas de fine tunning en el caso de MLP.

1.3. Base de datos

Las series de tiempo correspondientes a las curvas de luz son obtenidas desde el catálogo EROS. En particular, 9592 binarias eclipsantes (EB), 17836 RR Lyrae (RRL) y 2763 cefeidas (CEP) son utilizadas.

2. Metodología

2.1. Preprocesamiento

El enfoque seguido se basa principalmente en el procedimiento de obtención de características seguido por [2]. El preprocesamiento consistiría principalmente de la extracción de características de las series de tiempo de las curvas de luz. Tales características se obtendrían del siguiente modo: La primera frecuencia es entregada en forma de periodo en conjunto con la curva de luz. Ésta es utilizada para doblar la curva y dejarla en su representación fásica. Luego, un método de tratamiento del ruido propuesto en [1] fue aplicado a cada curva de luz, obteniendo resultados similares observado en la tercera imagen en la Figura 1, e; que es utilizado en un ajuste armónico con la siguiente forma:

$$y(t) = \sum_{j=1}^{4} (a_j sin(2\pi f_1 jt) + b_j cos(2\pi f_1 jt) + b_0$$

con y(t) la magnitud en función del tiempo. Luego, esta curva es restada de la serie de tiempo y un periodograma Lomb-Scargle es computado sobre la curva, eligiendo la frecuencia con mayor significancia. El mismo procedimiento es realizado hasta que tres frecuencias son encontradas (incluyendo la asociada al periodo de la curva). Finalmente, estas tres frecuencias son utilizadas para generar un mejor ajuste armónico a la serie de tiempo original:

$$y(t) = \sum_{i=1}^{3} \sum_{j=1}^{4} (a_{ij} sin(2\pi f_i jt) + b_{ij} cos(2\pi f_i jt) + b_0$$

con b_0 la magnitud media de la curva de luz. Las frecuencias f_i y los coeficiente a_{ij} y b_{ij} nos entregarían una descripción relativamente buena de las curvas de luz. Sin embargo, haría falta una

normalización para que los coeficientes de Fourier sena invariantes a traslaciones de tiempo, por lo que definimos $A_{ij} = \sqrt{a_{ij}^2 + b_{ij}^2}$ y $PH_{ij} = arctan(b_{ij}, a_{ij})$. Con esto, A_{ij} es invariante a traslaciones temporales, pero PH_{ij} no. Para ello primero hacemos que PH_{11} sea 0 y cambiamos los otros PH_{ij} correspondientemente:

$$PH'_{ij} = arctan(b_{ij}, a_{ij}) - arctan(b_{11}, a_{11})$$

con $PH'_{11} = 0$ (el primer armónico de f_1 ha sido elegido como referencia). Notar que estas nuevas fases pueden tener valores entre $-\infty$ e ∞. Por ello definimos:

$$PH''_{ij} = arctan(sin(PH'_{ij}, cos(PH'_{ij}))$$

con lo que finalmente tenemos una representación invariante a traslaciones temporales:

$$y(t) = \sum_{i=1}^{3} \sum_{j=1}^{4} A_{ij} sin(2\pi f_i j t) + PH''_{ij}) + b_0$$

Con esto tenemos 26 parámetros: la pendiente 3 frecuencias, 12 amplitudes, 11 fases (PH_{11} es siempre 0, por lo que puede ser descartada).

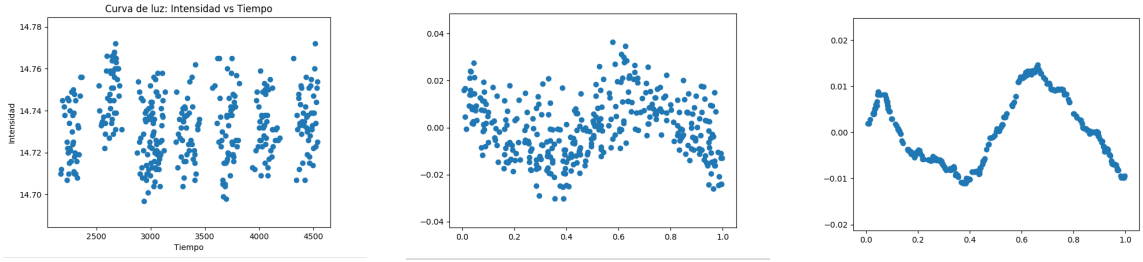


Figura 1: Ajuste de series de tiempo a fase de estrella cefeida

2.2. Métodos de clasificación

Por un lado, los MLP son conocidos clasificadores que tienden a requerir de una alta tasa ejemplos/parámetros para funcionar adecuadamente, pero si esto se cumple, la red tiende a funcionar de manera robusta y efectiva frente a la complejidad y el ruido. En el caso particular enfrentado, la cantidad de parámetros es considerablemente menor a la cantidad de ejemplos disponibles, por lo que las condiciones se cumplen.

En el caso de RF, este es un algoritmo que ya ha sido utilizado previamente con éxito en la clasificación de curvas de luz ([2] y[3]), por lo que es un buen candidato para la comparación deseada.

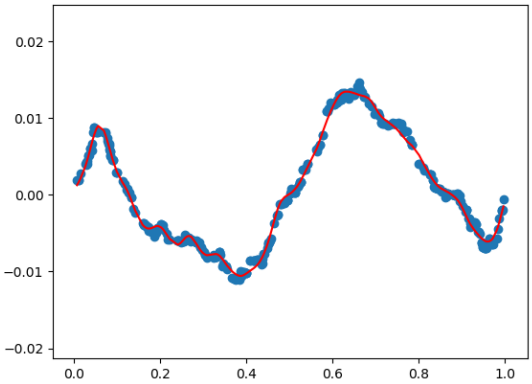


Figura 2: Fit de fourier final a curva de luz de estrella cefeida

2.3. Estructura e hiperparámetros

Las salidas de los clasificadores consistió en las distintas clases de Cefeidas (1,12,F) RR Lyraes (RRab,RRc,RRd,RRe) y eclipses binarias (ED, ED, ELL_EC, ED_VAR). En el caso del MLP, la función objetivo fue entropía cruzada y Adam el principio de optimización, mientras que en el caso de RF fueron utilizados los parámetros por defecto entregados por la librería scikit learn, es decir, 10 árboles de decisión y gini como el criterio para comparar los splits posibles (en vez de entropía).

El procesamiento y entrenamiento de los métodos de clasificación fueron realizados en Python mediante las bibliotecas scipy (periodogram, random forest) y tensorflow (mlp).

2.4. Resultados

Luego de obtener los 26 parámetros de cada ejemplo de curva de luz, se trató el tema del desbalance existente entre las clases: Cep: 2764, RRL: 17837, ECL: 9593. Dado que las RRL poseían la mayor cantidad de muestras, las otras dos clases fueron aumentadas (hasta alcanzar los 17837) con pequeñas variaciones aleatorias uniformes de su amplitud, con un aumento de ella entre 0 y 8 % (debido a una disminución aproximada de ella en un 4 % al aplicar el método de disminución de ruido).

En las Tabla 1 pueden observarse los resultados de la 'ten-fold´ validación cruzada realizada al utilizar Random Forest. Como es de esperar, los mejores resultados fueron obtenidos al utilizar los 26 características. Sin embargo, se puede aprecier que la ganancia desde la utilización de 1 a 3 (las tres frecuencias) se obtuvo una ganancia sustancialmente mayor que la ganancia desde 3 a 26. Por ello es que se realizó una comparación entre las significancias de las características, obtenindo que solo 11 de ellas combinadas sumaban el 90 % (Tabla 2). Puede observarse particularmente que las características asociadas a la tercera frecuencia tienden a tener un mayor impacto que la segunda, y que ninguna de las fases estaría presente en las características de mayor significancia.

Parámetros	Training	Testing	F1 Score
1	96.17 ± 0.09	66.30 ± 1.85	66.30 ± 1.80
3	98.85 ± 0.09	84.67 ±1.46	84.56 ± 1.48
26	99.82 ± 0.03	95.06 ± 1.15	95.05 ± 1.16

Tabla 1: Resultados Validación cruzada Random Forest, 3 clases, por número de parámetros

Característica	%	A14	4.2
F1	26.2	A21	3.9
A11	16.7	A33	3.8
F3	12.4	A12	3.8
F2	7.6	A34	2.7
A31	4.8	A13	2.6

Tabla 2: Significancia de 11 principales características (90 % significancia)

Posteriormente se procedió a realizar los experimentos asociados a MLP. Primero se realizaron algunos experimentos para ajustar la cantidad de épocas requeridas, observándose que para 400 épocas lograba ser suficiente para todas las configuraciones probadas. Para ello se utilizó un 80 % de training y 10 % tanto para validación y testing. En la Figura 3 pueden observarse algunos resultados. Puede apreciarse un overfitting casi inexistente, lo que podría deberse al pequeño tamaño de las redes utilizadas y a propiedades de las características seleccionadas para la clasificación, una gran efectividad, quizás debido a que éstas representarían a verdaderas características definitorias de las distintas clases de estrellas.

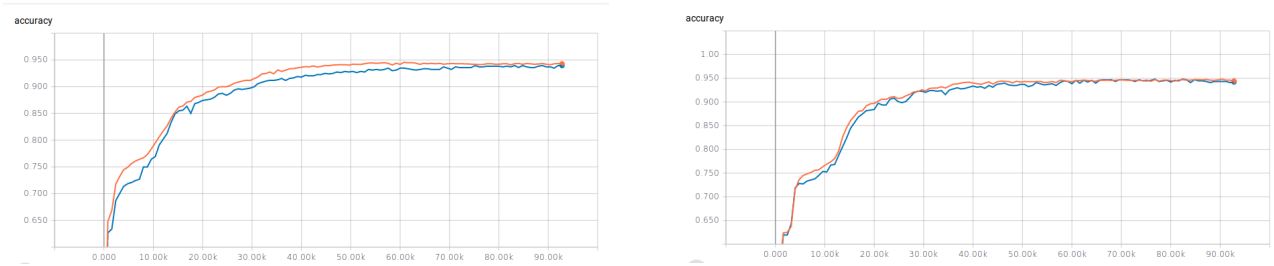


Figura 3: Resultados entrenamiento MLP 3 clases

Luego, se realizaron sucesivas validaciones cruzadas, primero para obtener un minibatch adecuado, y luego una buena configuración de capas para el MLP. Para llevar a cabo la segunda serie de validaciones cruzadas, se utilizaron heurísticas propuestas en [4], las que aconsejan el uso de una única capa escondida, además de partir con una cantidad de neuronas $H = 2/3(O + I)$, con O la cantidad de clases (neuronas en capa de salida) e I la cantidad de características (neuronas en capa de entrada). Luego, se consideró la cantidad de características significativas, a modo de conjetura de si ellas serían un mejor estimador para obtener la cantidad adecuada de H (neuronas en capa escondida). Para ello I se igualó a 11, con lo que se obtuvo un H = 9.33. Finalmente se halló que 10 neuronas presentaban los mejores resultados, a pesar de que no con demasiada contundencia.

Minibatch	Training	Testing
16	94.14 ± 0.53	93.96 ± 1.14
32	94.15 ± 0.30	94.02 ± 1.24
64	92.81 ± 1.57	92.31 ± 2.01

Tabla 3: Resultados MLP validación cruzada y 3 clases por tamaño minibatch

Neuronas (HL)	Training	Testing
9	94.14 ± 0.53	93.96 ± 1.14
10	94.15 ± 0.30	94.02 ± 1.24
11	94.41 ± 0.41	93.82 ± 1.27
19	94.33 ± 0.24	93.90 ± 1.10

Tabla 4: Resultados MLP validación cruzada y 3 clases por número de neuronas en capa escondida

Luego, se procedió a analizar las subclases de estrellas variables, para así poder tratar el desbalance en esta segunda etapa del proyecto. En la Tabla 5 se puede apreciar que 9 de las subclases cumplen con el criterio relativamente arbitrario elegido de una cantidad de muestras mayor a 500, para lograr una adecuado entrenamiento de los dos modelos de clasificación. Todas las subclases fueron aumentadas hasta alcanzar la cantidad de 1911, cantidad mayor observada entre las subclases.

CEP		ECL		RRL	
F	1494	ED	4931	RRab	12911
1	1032	ESD	3354	RRc	3526
12	167	EC	751	RRe	719
F1	52	ED_ESD	329	RRd	681
2	12	ELL_EC	153		
123	3	ED_VAR	60		
13	2	ESD_DPV	12		
F12	2	ED_TEB	3		

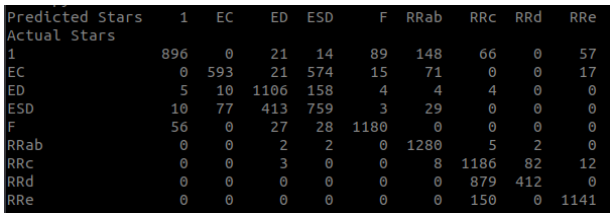
Tabla 5: Ejemplos por subclase de estrellas periódicas

En las Tabla 6 pueden observarse los resultados de entrenamiento y testing al utilizar Random Forest con las 9 subclases seleccionadas. El testing fue realizado en base a un 10 % de de los ejemplos no aumentados (fueron seleccionados previamente a la aumentación), y posteriormente aumentados para constituir un 10 % de los datos aumentados. Esto se verificó como muy importante, debido a que la aumentación previa a la separación generó resultados artificialmente buenos.

Nuevamente como era de esperar, los mejores resultados fueron obtenidos al utilizar los 26 ca-

racterísticas. Aunque la accuracy es inferior, ésta es no tan lejana a la alcanzada en el estado del arte ([3], 78 %), a pesar de que en tal caso fueron tratadas una mayor cantidad clases, pero también utilizadas una mayor cantidad de características.

Luego, en la Figura 4 puede apreciarse la matriz de confusión para las 9 subclases. En particular las subclases de binarias eclipsantes EC y ESD con la subclases de RRLyrae, RRd resultaron mayormente conflictivas. Las dos primeras parecen tener conflictos mutuamente, mientras que RRd tiende a confundirse con la subclase RRc.



Predicted Stars / Actual Stars	1	EC	ED	ESD	F	RRab	RRc	RRd	RRe
1	896	0	21	14	89	148	66	0	57
EC	0	593	21	574	15	71	0	0	17
ED	5	10	1106	158	4	4	4	0	0
ESD	10	77	413	759	3	29	0	0	0
F	56	0	27	28	1180	0	0	0	0
RRab	0	0	2	2	0	1280	5	2	0
RRc	0	0	3	0	0	8	1186	82	12
RRd	0	0	0	0	0	0	879	412	0
RRe	0	0	0	0	0	0	150	0	1141

Figura 4: Matriz de confusión para 9 subclases, obtenida desde Random Forest

Parámetros	Training	Testing	F1 Score
1	99.75 ± 0.01	46.51 ± 0.20	44.32 ± 0.25
3	99.96 ± 0.00	53.77 ± 0.44	52.13 ± 0.59
26	99.99 ± 0.00	72.63 ± 0.76	71.69 ± 0.82

Tabla 6: Resultados Random Forest, 9 clases, por número de parámetros

Finalmente, en la Tabla 7 pueden observarse los resultados de entrenar el MLP con los datos separados de las 9 clases. Utilizando el criterio H = 2/3(I+O), se obtuvo 23.33, resultado que al aproximar (23) y probar, se obtuvo un resultado aproximadamente igual (ligeramente superior) al obtenido por random Forest. Cabe señalar que los entrenamientos realizados con MLP obtuvieron una varianza superior debido a que solo 2 experimentos por configuración fueron realizados. Por ello ellos son un tanto menos confiables a los obtenidos para Random Forest.

3. Conclusiones

Los principales aprendizajes obtenidos estuvieron asociadoas a la relevancia de la selección de características: clave en alcanzar una buena clasificación. Por otro lado, se percibió que la utilización

Neuronas (HL)	Training	Testing
10	67.81 ± 0.37	66.80 ± 0.32
13	65.31 ± 3.39	65.47 ± 3.38
22	65.05 ± 2.26	64.42 ± 2.31
23	74.00 ± 2.08	72.70 ± 2.10

Tabla 7: Resultados MLP, 9 clases, por número de neruonas en capa escondida

de heurísticas en la realización de fine tunning sobre los hiperparámetrs resulta ser muy efectivo, reduciendo el tiempo de pruebas y alcanzando mejores resultados. Finalemente, se aprendió que la aumentación de los ejemplos debe ser realizada DESPUÉS de separar en Training, Validación y Testing. Adicionalmente, se logró verificar parcialmente que modelos distintos correctamente entrenados deberían alcanzar resultados similares en un problema de clasificación común.

También algunas conjeturas han sido propuestas para ser verificadas. En particular, que el bajo overfitting en MLP se debería al pequeño tamaño de la red y a la buena elección de los parámetros. Que el considerar cantidad de características críticas (parte de las que sumarían el 90 % de la significancia) para la elección de neuronas escondidas sería una variable importante a considerar para una heurística de selección de la cantidad de neuronas en la capa escondida. Y finalmente, que la utilización de un tipo de red neuronal distinta, como una red convolucional, no lograría mejorar significativamente los resultados obtenidos. Futuros experimentos permitirían verificar o refutar tales conjeturas.

4. Referencias

- P. Protopapas, J.M. Giammarco, L. Faccioli, M.F. Struble, R.Dave, C. Alcock, "Finding outlier light curves in catalogs of periodic variable stars", Monthly Notices of Royal Astronomical Society, Vol 369, pp 677-696, 200

- J. Debosscher, L. M. Sarro , C. Aerts, J. Cuypers, B. Vandenbussche, R. Garrido, and E. Solano, Automated supervised classification of variable stars, Astronomy & Astrophysics, 475.3 (2007): 1159-1183

- J. W. Richards, et al. "On machine-learned classification of variable stars with sparse and noisy time-series data." *The Astrophysical Journal*, 733.1 (2011): 10

- Panchal, Ganatra, et al. "Behaviour Analysis of Multilayer Perceptrons with Multiple Hidden Neurons and Hidden Layers" IJCTE 3.2 (2011): 1793-8201.