Predicción y Trabajo Final

Desarrollo de Sistemas Inteligentes

Alberto Velasco Mata
Alberto.Velasco1@alu.uclm.es
Escuela Superior de Informática
Ciudad Real

Antonio Pulido Hernández
Antonio.Pulido@alu.uclm.es
Escuela Superior de Informática
Ciudad Real

RESUMEN

In this document an analysis of the data related with the 2020 European Women's Handball Championship will be presented. The statistics about each player involved in the championship and the results of each game were used to design a model that predicts the results of hypothetical games between the different teams that participated in the tournament.[1]

1. INTRODUCCIÓN

En este documento se expondrán los análisis, pruebas y modelos que se han llevado a cabo para conseguir una predicción de un posible resultado para un partido del campeonato Europeo de Balonmano Femenino. Se detallarán las decisiones tomadas teniendo en cuenta los datos tomados como base, así como la elección de las distintas técnicas empleadas y los resultados obtenidos. Seguidamente, se hará una reflexión sobre los resultados obtenidos tanto en el trabajo anterior de *clustering* como en la predicción expuesta, y se propondrán posibles mejoras a los mismos.

2. PREDICCIÓN

En este trabajo se pretende utilizar algoritmos de predicción partiendo de los datos correspondientes a las estadísticas del Europeo Femenino de Balonmano del año 2020 en los que se puede encontrar información relevante de las jugadoras en cada uno de los partidos disputados.

2.1. Conjuntos de datos

En primer lugar, necesitamos definir con qué datos se va a trabajar, ya que esto será un factor determinante que afectará al modelo. El conjunto de datos del que disponíamos contenía únicamente las variables de cada jugadora en cada partido del Campeonato. Aunque sí es cierto que se podría haber utilizado la suma total de los puntos de las jugadoras en un mismo equipo y partido para obtener el marcador final, hemos optado por tomar los datos directamente de la fuente proporcionada en el enunciado[2]. Estos datos han sido definidos en un archivo en formato YAML, para facilitar su lectura y procesado posteriormente.

El hándicap principal que fue notable desde el principio es el reducido volumen de datos del que se disponía, habiendo únicamente 24 partidos en la primera fase del Europeo, y 23 partidos en el resto de fases. Esto ha dado lugar a considerar técnicas de *data augmentation* que se explicarán posteriormente.

Se consideró también la posibilidad de descargar estadísticas adicionales de años anteriores para incrementar el volumen de datos. Sin embargo, tras consultar varias fuentes, se descartó esta opción debido a que las fuentes oficiales han quitado el acceso público a los campeonatos anteriores, y la escasa información que se pudo encontrar era muy reducida y en formatos que no estaban pensados para su procesamiento automático.

2.2. Definición del modelo

A continuación, se ha procedido a definir conceptualmente el modelo que se va a implementar. Se parte del supuesto de que para predecir el resultado de un partido será necesario conocer las características que definan o representen a los dos equipos que participen en el mismo. De esta manera, el conjunto de datos procesado estará compuesto por la unión de un vector de características correspondiente al primer equipo, un vector de características correspondiente al segundo equipo, y un valor adicional que indique cuál de los dos resultó ganador. Este valor será el resultado de la predicción, y será un valor categórico ('empate', 'primero' o 'segundo'), por lo que estamos ante un problema de clasificación.

Teniendo en cuenta esta propuesta, se han considerado tres posibles opciones para la selección de características representativas de los equipos:

- **Caso 1.** Utilizar directamente los resultados del análisis de componentes principales que se realizó en el trabajo anterior, el cuál estaba enfocado en caracterizar los equipos. Además, se incluiría el *cluster* obtenido en el mismo. Se considerarían, por tanto, tres variables para cada uno de los equipos del partido, y podrían ser una buena representación de los equipos debido a que el análisis cluster fue validado con el experto.
- **Caso 2.** Utilizar de nuevo todas las variables útiles que se obtuvieron tras el análisis inicial del trabajo anterior, junto con el cluster obtenido. Esta opción podría, potencialmente, dejar elegir al modelo de predicción las variables más influyentes o significativas para la obtención del resultado.
- **Caso 3.** Partir del caso anterior y realizar un nuevo análisis de componentes principales. Este análisis sería similar al que se realizó en el trabajo anterior, con la diferencia de que se incluiría el clúster como información adicional.

Las técnicas que se ha decidido utilizar han sido **KNN** y **SVM**, debido a lo siguiente:

- Todas las características consideradas son valores numéricos, por lo que el cálculo de la distancia no supone un problema.
- En los casos 1 y 3, la dimensionalidad del problema será bastante reducida, por lo que no debería suponer un problema para el método KNN.
- La parametrización es sencilla y puede ajustarse fácilmente antes de utilizarlos para clasificar los resultados de los partidos.

[1] El código se encuentra disponible en: https://github.com/AlbertoVelascoMata/course.dsi

	KNN	SVM
Caso 1 (PCA & *cluster*)	**87.5 %**	**87.5 %**
Caso 2 (*features* & *cluster*)	79.17 %	83.34 %
Caso 3 (PCA de *features* & *cluster*)	83.34 %	83.34 %

Tabla 1: Porcentajes de acierto conseguidos con cada combinación de los casos y modelos propuestos.

- El conjunto de datos es pequeño, por lo que el cálculo de las distancias en el método KNN no tendrá un coste elevado.
- En el caso 2, el número de características podría ser superior a la cantidad de muestras disponibles. SVM mantiene su efectividad en estas situaciones por lo que podría ser un buen candidato.

2.3. Validación de los modelos

Como se ha mencionado anteriormente, se ha decidido utilizar inicialmente los partidos correspondientes a la primera fase como validación de los modelos, estando reservados únicamente para la comprobación de la bondad de los mismos. Por lo tanto, los partidos que se han utilizado para el entrenamiento han sido los 23 correspondientes a la segunda fase y fase final del Campeonato.

Para incrementar la cantidad de datos disponibles y añadir robustez al modelo, se ha empleado ***data augmentation***, duplicando el número de partidos disponibles en el entrenamiento. Para ello, y teniendo en cuenta que cada muestra está formada por las características de ambos partidos y el resultado, se han invertido los equipos. De esta manera, por ejemplo, un partido entre Dinamarca y Rusia en el que gana el primer equipo (Dinamarca), se considera equivalente a un partido entre Rusia y Dinamarca en el que gana el segundo equipo (Dinamarca). Esto ha permitido disponer de 46 muestras en total para el entrenamiento, así como **evitar que el orden de las mismas afecte al modelo implementado** (ya que si todas las muestras estuviesen representadas de manera que el equipo ganador apareciese primero, el modelo aprendería que la salida siempre será "primero", porque no dispondría de ejemplos en el entrenamiento que indicasen que el resultado es "segundo").

Se ha procedido al entrenamiento y validación de los modelos KNN y SVM con las tres situaciones que se mencionaron en la Sección 2.2. En cada uno de los casos, se ha hecho un **estudio empírico de la parametrización óptima**, el cuál puede verse en el código. La Tabla 1 muestra los resultados obtenidos, siendo interesante destacar que el método SVM proporciona un mejor resultado que KNN en el caso 2, donde la dimensionalidad era elevada tal y como se ha comentado en la justificación de la elección de estos algoritmos.

El mejor resultado obtenido ha conseguido un **porcentaje de acierto del 87.5 %** en los partidos del conjunto de validación, correspondiéndose con el método KNN parametrizado con una k = 4 (es decir, tiene en cuenta a los cuatro vecinos más cercanos) y dándole un peso uniforme a los vecinos (todos pesan lo mismo, en lugar de depender de la distancia). La matriz de confusión puede verse en la Figura 1. Lo más importante a destacar es el hecho de que el modelo nunca predice un empate. Esto se debe al desbalanceo que existe en los datos de entrenamiento, habiendo solamente un partido en el que los equipos empatasen. El método SVM ha

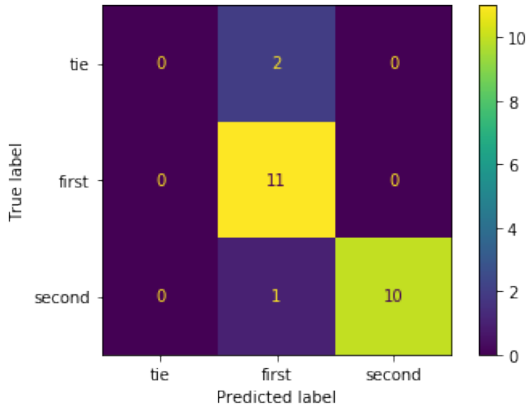

Figura 1: Matriz de confusión con el método KNN y las variables definidas en el caso 1

teamA	teamB	result
ESP	HUN	1 (ESP)
CRO	ESP	1 (CRO)
DEN	NED	1 (DEN)
CZE	POL	1 (CZE)
FRA	GER	1 (GER)

Tabla 2: Pronóstico para partidos no jugados en el Campeonato.

dado el mismo resultado con estos datos, pero se ha decidido por simplicidad utilizar el modelo KNN.

2.4. Pronóstico

Una vez escogido el modelo con el método KNN parametrizado con una k = 4 se ha procedido a aplicarlo para intentar calcular el resultado de un conjunto de partidos entre equipos que participaron en el campeonato, pero que no llegaron a jugar entre ellos, obteniendo los resultados expuestos en la Tabla 2. En la tabla se puede observar que los 5 equipos que juegan como local son los que ganan el partido. Para comprobar la consistencia y robustez del modelo, se ha probado a invertir la situación de local y visitante, obteniendo como resultado que los 5 equipos visitantes ganaban.

A pesar de que no se pueden validar estos resultados debido a que no han tenido lugar estos partidos, se han comparado respecto a la clasificación final de los equipos en el Campeonato para comprobar que tienen sentido. Se observa que los que ganarían según la predicción son los que han quedado mejor clasificados en el torneo. Por lo tanto, se puede afirmar que el pronóstico obtenido con el modelo desarrollado es bastante fiable y se asemeja a la realidad.

2.5. Conclusiones

En general, podríamos decir que los resultados obtenidos se ajustan bastante a la realidad obteniendo con el mejor modelo un porcentaje de acierto del 87.5 % en los partidos del conjunto de validación.

Sin embargo, se han encontrado algunos problemas durante la realización del trabajo. En específico, el pequeño tamaño del conjunto de datos ha significado que se utilicen únicamente 23 partidos para entrenar el modelo. Este factor, junto con el hecho de que los datos estaban desbalanceados (solo hay una muestra en el conjunto de entrenamiento en la que los equipos empaten el partido), es una limitación que ha tenido que compensarse con técnicas de aumento de datos para conseguir un resultado funcional.

En conclusión, aunque los resultados obtenidos son bastante fiables habiéndose validado con un alto porcentaje de acierto, los resultados hubieran sido más precisos con un conjunto de datos con más partidos para entrenar los modelos. Esto significa que los algoritmos y modelos aplicados no han mostrado todo su potencial en este caso de estudio y podrían potencialmente ser más precisos si se usasen los datos de otros torneos como los campeonatos de los años anteriores.

3. TRABAJO FINAL

3.1. Introducción

En el trabajo final se pretende hacer una reflexión de los algoritmos utilizados en los trabajos de clustering y predicción, la adecuación del algoritmo seleccionado para analizar las estadísticas y otros posibles algoritmos que podrían haber funcionado mejor que los utilizados.

3.2. Clustering

En los trabajos de clustering se han usado los algoritmos C-means, Algoritmos jerárquicos, Clustering basado en densidad, Expectation-Maximization y Mapas Autoorganizados siendo el seleccionado para nuestro trabajo el algoritmo Expectation-Maximization [1]. En este algoritmo cada cluster es representado por una distribución paramétrica, y normalmente se utilizan funciones gaussianas, como en el caso expuesto. Para cada elemento se calcula la probabilidad de que sea generado por cada gaussiana, y habitualmente se considera que pertenece al cluster con mayor probabilidad o grado de pertenencia.

La principal ventaja que se ha detectado en este algoritmo es la asignación probabilística del cluster, lo cual hace posible entender el grado de pertenencia a varios clusters. Consideramos que esta característica nos ha favorecido especialmente al agrupar las jugadoras, ya que, como se representó visualmente en el trabajo anterior, los conglomerados estaban muy solapados entre sí. De esta manera, la asignación probabilística no se ve tan empeorada por ese solapamiento como otros algoritmos discretos como podría ser k-Means.

Además, el algoritmo Expectation-Maximization permite manejar clusters con diferentes tamaños de conjuntos de datos y varianzas, siendo más adecuado que, por ejemplo, k-Means en este problema concreto.

Sin embargo, también tenemos que destacar el que consideramos que ha sido el principal inconveniente del algoritmo: el problema de la inicialización. En la práctica hemos observado la gran variabilidad de resultados a la que daba lugar este algoritmo. Se detectó este problema inicialmente al agrupar a las jugadoras, aunque en ese caso parecía converger a dos posibles soluciones y la mejor fue seleccionada con ayuda del experto, teniendo en cuenta el criterio que proporcionó indicando que tenía más sentido que el resultado se correspondiese con la posición de juego.

Al aplicar el clustering en los equipos, este problema se intensificó notablemente. El hecho de que hubiese una cantidad tan limitada de muestras (solo 16 equipos) dio lugar a que la inicialización aleatoria cambiase considerablemente la salida del algoritmo. De nuevo, si no hubiese sido por el conocimiento experto y la disponibilidad de la clasificación final de los equipos en el Campeonato, no habría sido posible conseguir un resultado útil y que se acercase a la realidad.

Teniendo todo esto en cuenta, consideramos que la técnica asignada era adecuada para la agrupación de las jugadoras debido al grado de pertenencia que proporciona, mientras que para la agrupación de los equipos podría haber sido más adecuada otra técnica que no dependiese tanto de la inicialización.

Es importante destacar que, aunque el algoritmo no fuese óptimo para la agrupación de los equipos, el resultado propuesto parecía asemejarse a la realidad, por lo que hemos mantenido ese resultado al realizar la predicción.

3.3. Predicción

En el trabajo de predicción se han usado los algoritmos KNN y SVM para predecir los resultados de los partidos del Europeo Femenino de Balonmano de 2020. Se ha comprobado que se obtiene un mejor resultado utilizando el análisis cluster previo, en lugar de utilizar todas las variables. Se podría haber considerado utilizar un árbol de decisión. Sin embargo, lo hemos descartado por tres motivos:

- No se aprovecharía la ventaja que proporcionan los árboles en cuanto a la interpretabilidad de los resultados si se hubiese aplicado al caso 1 (Sección 2.2), ya que ahí se utilizaban las componentes principales junto con el cluster obtenido. Solo habríamos podido, como mucho, interpretar qué grupos de equipos ganan o pierden más, y esa información estaría entremezclada con el resultado del análisis de componentes principales.
- Si se hubiese utilizado el árbol de decisión y el conjunto completo de características (como en el caso 2 que se definió en la Sección 2.2), tendríamos un árbol de mayor tamaño, complicando su interpretabilidad.
- En cualquier caso, los árboles tienden al sobreaprendizaje. Teniendo en cuenta la cantidad tan limitada de datos de los que disponemos, sería muy propenso al overfitting.

A pesar de que los resultados obtenidos son bastante buenos, sí que consideramos que podría haberse obtenido un modelo de predicción más potente. Hay distintas alternativas (incluyendo combinaciones de ellas) que podrían dar lugar a ello:

- **Obtener un volumen de datos mayor.** Al fin y al cabo, estos algoritmos dependen mucho de los datos con los que son entrenados. La mayor limitación que hemos observado ha sido la escasez de muestras. Incluso cogiendo todas las muestras disponibles y aplicando la técnica de *data augmentation* mencionada previamente, siguen siendo poco. Por lo tanto, añadir más datos podría mejorar considerablemente el modelo.

- **Incluir datos que sean más representativos del modo de juego.** Las estadísticas proporcionadas en el conjunto de datos puede que sean útiles para definir un perfil para las jugadoras, pero no pensamos que sean las mejores para predecir el resultado de un partido. Probablemente serían necesarias más variables que indiquen cómo se comporta cada equipo a lo largo del partido (quizá algunos indicadores adicionales de agresividad, defensa, etc. definidos por un experto).
- **Balancear los datos.** Principalmente enfocándonos en los empates. Somos conscientes de que, idealmente, es una situación poco común, pero en los datos de entrenamiento solo había un partido que haya resultado en empate, por lo que prácticamente todos los modelos que hemos probado directamente descartan la opción de que pueda darse ese caso. El balanceo de qué partido gana (el que aparece primero o el que aparece segundo) se ha compensado con la técnica de *data augmentation*, ya que hay la misma cantidad de ambas situaciones.
- **Utilizar otros modelos**, quizá más potentes, que puedan interpretar la importancia de cada una de las variables. Teniendo en cuenta la dimensionalidad inicial de los datos, quizá una red neuronal podría dar lugar a mejores resultados.
- En la implementación que hemos realizado de la predicción, los vectores de características de cada equipo se obtenían del total de partidos del Campeonato debido a que eran la salida del análisis cluster y a que consideramos que el "modo de juego" de cada equipo se vería mejor representado teniendo en cuenta todos los partidos, con una visión global. Sin embargo, **podría considerarse el caso de que algún equipo pueda jugar distinto dependiendo del partido**. Esto da lugar a una posible mejora: tener en cuenta una caracterización de los equipos por cada partido en el entrenamiento del modelo. No obstante, al utilizar el modelo para predecir el resultado de un partido no jugado tendría que mantenerse esa caracterización global ya mencionada.
- Quizá podría haber sido interesante desarrollar un modelo de predicción que no tenga en cuenta al equipo rival, sino que simplemente según el modo de juego de un equipo en un partido concreto sea capaz de **predecir sus posibilidades de ganar**. Habría que indagar un poco más en esta idea ya que el equipo rival es un factor muy influyente, por lo que quizá el modelo no sería muy útil.

3.4. Otras técnicas

Teniendo en cuenta las propuestas anteriores y que no se dispone de mucho margen de acción con los datos como se comentó en la Sección 2.1, se ha decidido probar el uso de una red neuronal para mejorar los resultados. En parte la decisión se debe también al interés de los miembros del equipo en este tema, y a que no se han realizado muchos ejemplos prácticos en la asignatura. Además, como se ha mencionado previamente, un modelo de predicción basado en una red neuronal podría determinar qué variables son realmente

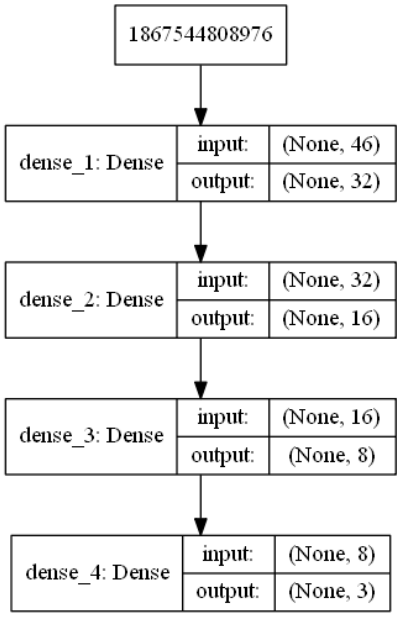

Figura 2: Visualización de las capas de la red neuronal definida para predecir el resultado del partido con muchas características en la entrada.

significativas para obtener el resultado, evitando la varianza explicada que se descarta al limitar el análisis de componentes principales a un número concreto.

Por tanto, se han realizado dos experimentos con redes neuronales:

- Un primer experimento, similar al Caso 2 que se mencionó en la Sección 2.2, en el que se utilizarán **todas las variables con información útil** para la predicción del resultado.
- Un segundo experimento, similar al Caso 1, debido a que las **tres variables por equipo** dieron el mejor resultado con el modelo de predicción basado en KNN.

En el primer experimento, con un mayor número de características, se ha definido una red neuronal completamente conectada con dos capas intermedias, como puede verse en la Figura 2. Con este modelo, se ha obtenido un porcentaje de acierto del 75 %, siendo inferior al que se obtuvo con KNN y SVM. Esto podría deberse, de nuevo, al volumen tan reducido de datos que se utilizan para entrenarlo, ya que no permite que la red pueda converger a una solución mejor. Podría deberse también al número de capas ocultas, pero incrementarlas con tan pocos datos daba lugar a que el entrenamiento se estancase casi desde el inicio, con un resultado peor.

En el segundo experimento, con únicamente seis características por cada partido (tres por equipo), se ha reducido el tamaño de las capas ocultas (pero no la cantidad), como puede verse en la Figura 3. Con este modelo se ha obtenido un porcentaje de acierto del 30 %, debido a que el modelo converge rápidamente a dar como salida que el ganador es el partido correspondiente al primer vector de

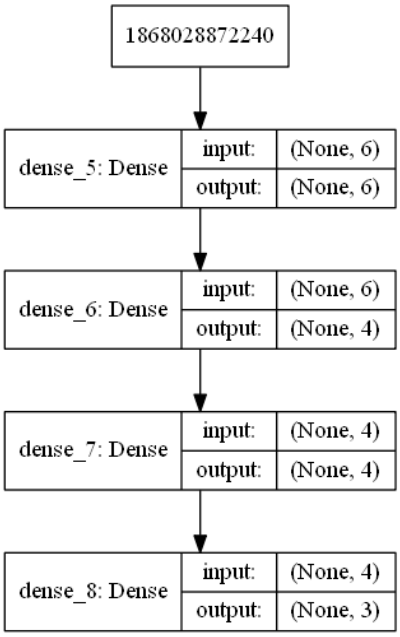

Figura 3: Visualización de las capas de la red neuronal definida para predecir el resultado del partido con tres características de cada equipo.

características. Podría deberse a que estamos limitando la información que le llega a la red neuronal en el entrenamiento, quitando las características originales de la entrada.

Como conclusión, a pesar de que no se ha conseguido mejorar el modelo (ya que los resultados son peores que los obtenidos inicialmente con KNN y SVM), sí que es interesante destacar que con la red neuronal pasa lo contrario a lo que ocurría en el caso inicial. Con KNN y SVM, el mejor resultado se obtenía tras haber reducido las variables y haber realizado un análisis cluster previo. Sin embargo, en el caso de la red neuronal, el mejor resultado se obtiene cuando esta toma como entrada el vector completo de características, lo cual concuerda con lo que se había expuesto anteriormente de que en este caso el modelo es el que está decidiendo qué variables son significativas para la predicción. Hay que considerar también, que debido al volumen de datos reducido, es posible que este modelo esté dando lugar al sobreaprendizaje.

REFERENCIAS

[1] T. K. Moon. 1996. The expectation-maximization algorithm. *IEEE Signal Processing Magazine* 13, 6 (1996), 47–60. https://doi.org/10.1109/79.543975

[2] Wikipedia. 2021. Campeonato Europeo de Balonmano Femenino de 2020 — Wikipedia, The Free Encyclopedia. http://es.wikipedia.org/w/index.php?title=Campeonato%20Europeo%20de%20Balonmano%20Femenino%20de%202020.