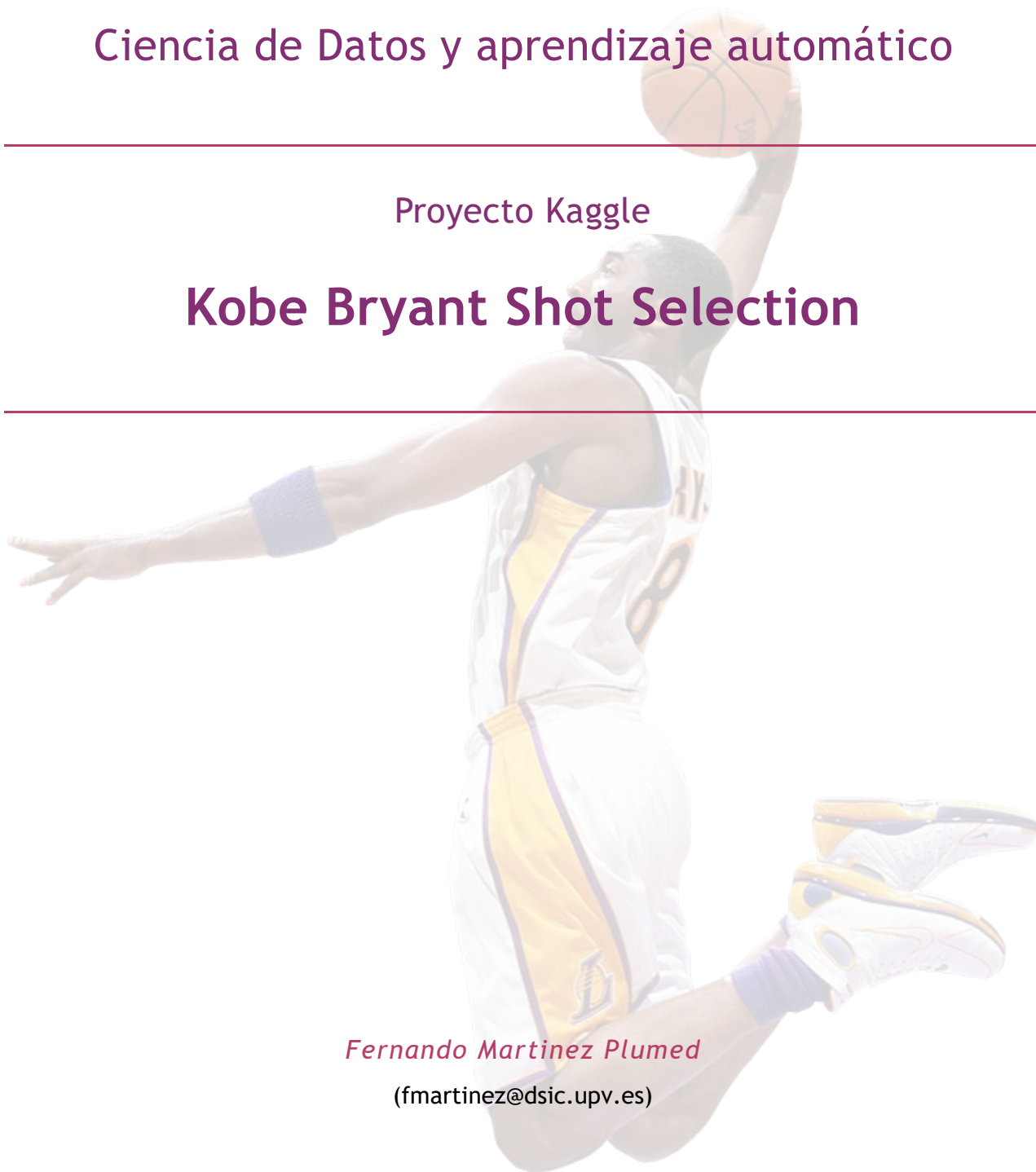


Ciencia de Datos y aprendizaje automático

Proyecto Kaggle

Kobe Bryant Shot Selection



Fernando Martinez Plumed
(fmartinez@dsic.upv.es)

MÁSTER UNIVERSITARIO EN INVESTIGACIÓN EN INTELIGENCIA ARTIFICIAL

UIMP

Universidad Internacional
Menéndez Pelayo



Asociación Española para la Inteligencia Artificial (**AEPIA**)

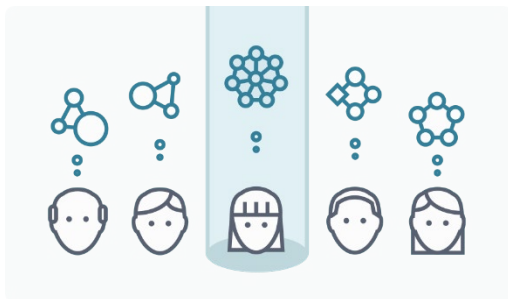
Describimos aquí el ejercicio evaluable a realizar en la plataforma Kaggle.

1. Kaggle

Kaggle (www.kaggle.com) es una plataforma web donde se reúnen miles de personas con interés o experiencia en el análisis de datos, ofreciendo la posibilidad de competir para resolver requisitos estratégicos que presentan los grandes datos de las empresas a cambio de dinero (o conocimiento). Empresas y compañías de todo el mundo exponen sus problemas y sus retos en esta plataforma y la comunidad de científicos de datos compete para crear las mejores soluciones y los mejores modelos teóricos.

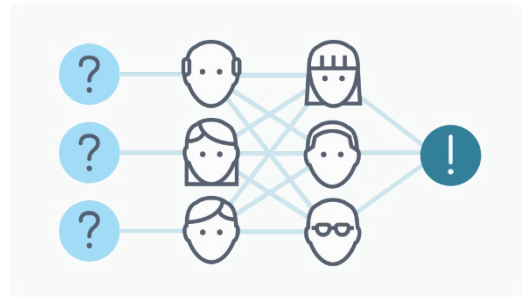
Empresas

Utilizan la comunidad de científicos más grande del mundo para consultar y resolver los problemas de negocio más complejos.



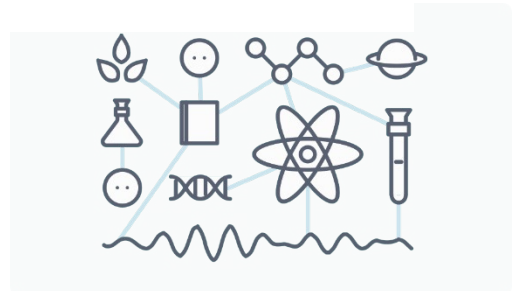
Contratación

Encuentran el mejor equipo por medio de competiciones a medida y la visibilidad del código utilizado por los candidatos, las colaboraciones llevadas a cabo y los resultados.



Investigación

Aceleran tu investigación alojando sus problemas de aprendizaje automático supervisado para el beneficio general.



En la plataforma se presenta cualquier tipo de problema que pueda encontrarse en los distintos campos del mundo real, tales como servicios financieros, energía, tecnología de la información, etc.

El enfoque “crowdsourcing” utilizado se debe a la existencia de una cantidad indefinida de posibles soluciones y estrategias que se pueden aplicar a un problema complejo de modelado predictivo donde no es posible saber con antelación la técnica o la estrategia que será más adecuada y más eficaz.

Fundada por el economista australiano Anthony Goldbloom, la inspiración para crear Kaggle proviene en parte de un concurso convocado por Netflix entre 2006 y 2009. La empresa de alquiler de películas ofrecía un millón de dólares al equipo que fuera capaz de mejorar la precisión de su software de recomendación de títulos en un 10 por ciento.

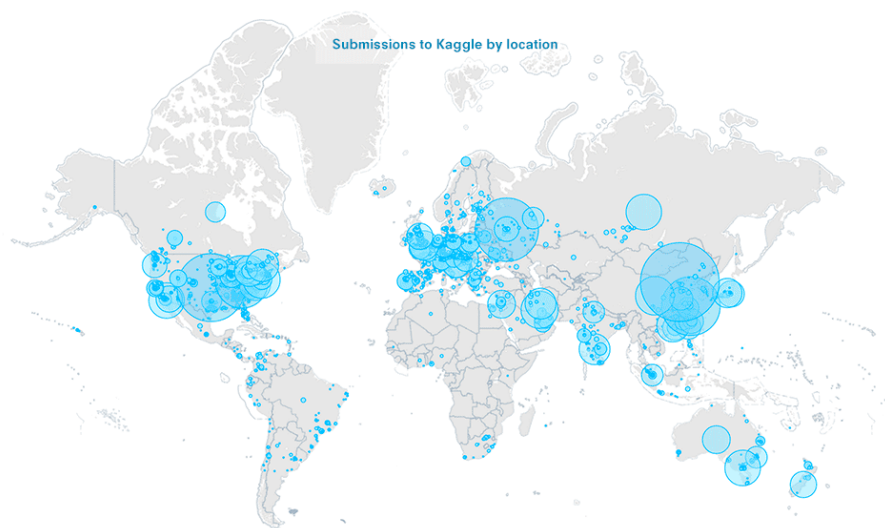


Figura 1. Numero de subidas a kaggle por localización.

1.1 Registro

El primer paso es registrarse en el sistema y crear manualmente una nueva cuenta de usuario (“Sign in” -> “manually create an account”).

Tanto el nombre de usuario (“User name”) como el nombre que se mostrará en la clasificación final (“Display name”) será “CDAA” seguido de tu nombre sin espacios (Ej.: “CDAA_JuanGomez”).

Una vez registrado (Figura 2), haz el *login* en el sistema y accede al apartado “Competitions” donde encontrarás una lista con todas las competiciones, tanto activas como completadas. Busca la competición titulada como “Kobe Bryant Shot Selection” (Figura 3) o accede a <https://www.kaggle.com/c/kobe-bryant-shot-selection>.

Figura 2. Registro en Kaggle

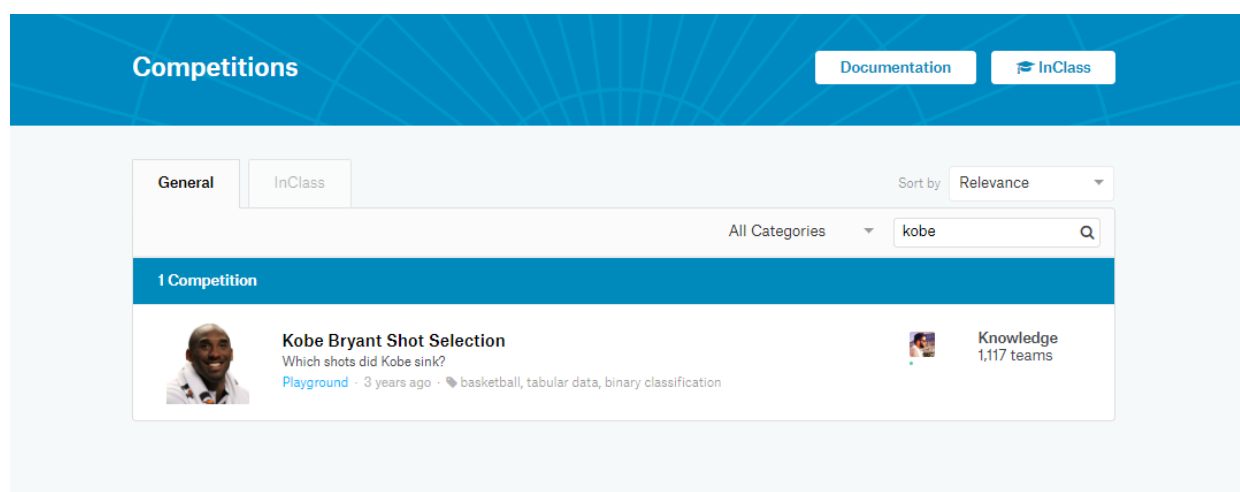


Figura 3. Acceso al apartado de Competiciones y a la Tarea correspondiente.

Para participar en la competición y descargar los datos del problema deberás aceptar las reglas de la misma.

2. Descripción del problema

Este proyecto busca llevar a cabo un proyecto completo de Ciencia de Datos que permita a los alumnos la oportunidad de demostrar las habilidades que han aprendido durante el curso. El primer requisito es que el conjunto de datos utilizado posea un número razonablemente grande de variables y observaciones para que los estudiantes tengan que ir más allá de la aplicación de un algoritmo simple de selección de variables para la construcción de un modelo final.

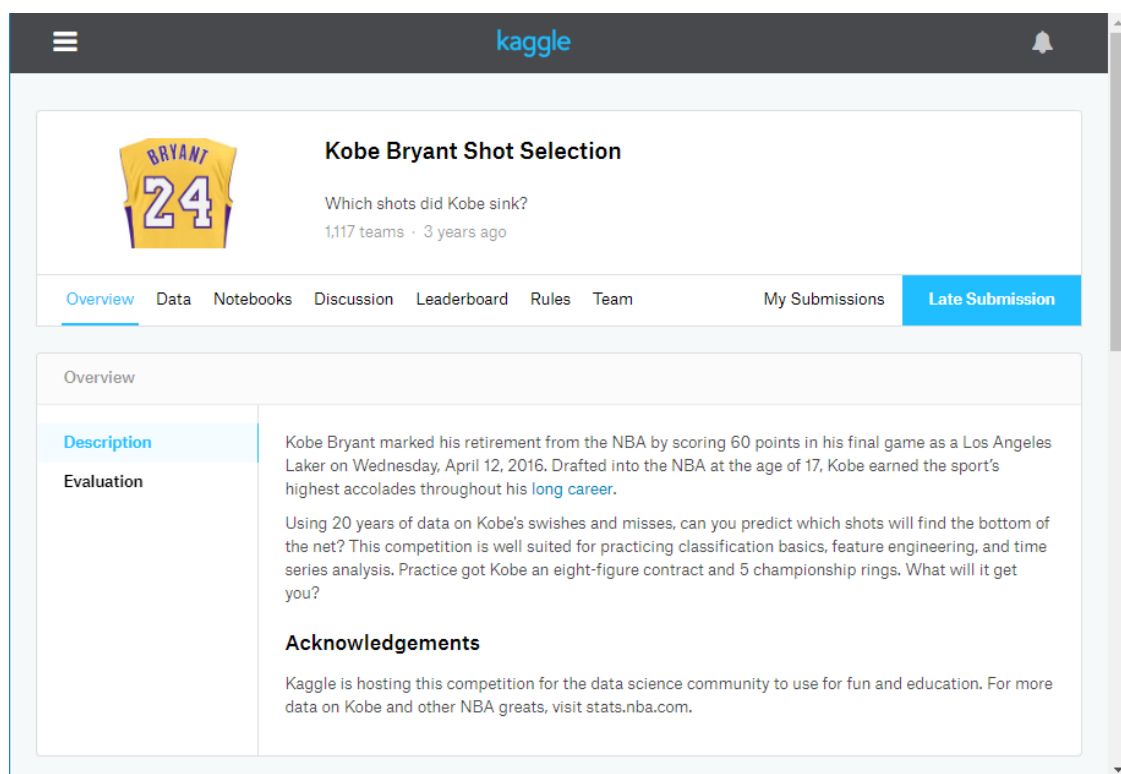


Figura 4. Detalles de la competición en kaggle.

Concretamente, para este proyecto se ha usado un conjunto de datos que describe los aciertos y fallos de lanzamientos a canasta del jugador de baloncesto Kobe Bryant¹ durante 20 años de carrera en la NBA. El conjunto de datos contiene 30697 observaciones y un gran número de variables explicativas (11 discretas y 14 numéricas). Estas 25 variables (incluyendo la clase a predecir) se centran en la descripción cualitativa y cuantitativa de multitud de aspectos de cada uno de los lanzamientos de Kobe Bryant.

¹ https://en.wikipedia.org/wiki/Kobe_Bryant

```
## Classes 'tbl_df', 'tbl' and 'data.frame': 30697 obs. of 25 variables:
## $ action_type : Factor w/ 57 levels "Alley Oop Dunk Shot",...: 27 27 27 27 6 27 28 27 27 42 ...
## $ combined_shot_type: Factor w/ 6 levels "Bank Shot","Dunk",...: 4 4 4 4 2 4 5 4 4 4 ...
## $ game_event_id : int 10 12 35 43 155 244 251 254 265 294 ...
## $ game_id : int 20000012 20000012 20000012 20000012 20000012 20000012 20000012 20000012 20000012 20000012 ...
## $ lat : num 34 34 33.9 33.9 34 ...
## $ loc_x : int 167 -157 -101 138 0 -145 0 1 -65 -33 ...
## $ loc_y : int 72 0 135 175 0 -11 0 28 108 125 ...
## $ lon : num -118 -118 -118 -118 -118 ...
## $ minutes_remaining : int 10 10 7 6 6 9 8 8 6 3 ...
## $ period : int 1 1 1 1 2 3 3 3 3 3 ...
## $ playoffs : int 0 0 0 0 0 0 0 0 0 0 ...
## $ season : Factor w/ 20 levels "1996-97","1997-98",...: 5 5 5 5 5 5 5 5 5 5 ...
## $ seconds_remaining : int 27 22 45 52 19 32 52 5 12 36 ...
## $ shot_distance : int 18 15 16 22 0 14 0 2 12 12 ...
## $ shot_made_flag : int NA 0 1 0 1 0 1 NA 1 0 ...
## $ shot_type : Factor w/ 2 levels "2PT Field Goal",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ shot_zone_area : Factor w/ 6 levels "Back Court(BC)",...: 6 4 3 5 2 4 2 2 4 2 ...
## $ shot_zone_basic : Factor w/ 7 levels "Above the Break 3",...: 5 5 5 5 6 5 6 6 3 3 ...
## $ shot_zone_range : Factor w/ 5 levels "16-24 ft.", "24+ ft.",...: 1 3 1 1 5 3 5 5 3 3 ...
## $ team_id : int 1610612747 1610612747 1610612747 1610612747 1610612747 1610612747 1610612747 1610612747 1610612747 1610612747 ...
## $ team_name : Factor w/ 1 level "Los Angeles Lakers": 1 1 1 1 1 1 1 1 1 1 ...
## $ game_date : Factor w/ 1559 levels "1996-11-03","1996-11-05",...: 311 311 311 311 311 311 311 311 311 311 ...
## $ matchup : Factor w/ 74 levels "LAL @ ATL","LAL @ BKN",...: 29 29 29 29 29 29 29 29 29 29 ...
## $ opponent : Factor w/ 33 levels "ATL","BKN","BOS",...: 26 26 26 26 26 26 26 26 26 26 ...
## $ shot_id : int 1 2 3 4 5 6 7 8 9 10 ...
```

En general, las variables cuantitativas hacen referencia a identificadores, posiciones específicas en la cancha, tiempo, distancias, tipos de partidos, etc. Por su parte, las variables discretas describen tipos de lanzamientos, temporadas, zonas en la cancha, oponentes, encuentros, etc. El alumno debe comprender y analizar cada una de estas variables.

La tarea del alumno es predecir si los lanzamientos a canastas de Kobe Bryant entraron o no en el aro ("shot_made_flag"). Del conjunto de datos se han eliminado 5000 valores de este atributo (representados como valores faltantes en el archivo csv). Estos datos serán el conjunto de prueba sobre el cual se realizará la predicción.

EL PROBLEMA QUE SE PRESENTA EN ESTA COMPETICIÓN RADICA, POR TANTO EN PREDECIR LOS LANZAMIENTOS ACERTADOS

Para comenzar a trabajar, en la pestaña "Data" encontraremos los ficheros **data.csv** y **sample_submission.csv** que cargaremos después en R. El conjunto de datos de (**data.csv**) consta de 30697 instancias (25697 + 5000) que tendremos que separar en train/test, tal y como se ya se ha comentado. El fichero **sample_submission.csv** proporciona un formato de muestra para el envío de resultados en Kaggle.



Data (721 KB)		API	kaggle competitions download -c kobe-bryant-shot...	?	Download All	✕
Data Sources		About this file		Columns		
	data.csv	30.7k x 25		A action_type		
	sample_submission....	5000 x 2		A combined_shot_type		
				# game_event_id		
				# game_id		
				# lat		
				# loc_x		
				# loc_y		
				# lon		
				# minutes_remaining		

Figura 5. Datos y scripts

Por otra parte, en la pestaña de datos nos encontramos con la siguiente advertencia:

To avoid **leakage**, your method should only train on events that occurred prior to the shot for which you are predicting! Since this is a playground competition with public answers, it's up to you to abide by this rule.

El alumno **no debe tenerla en cuenta** (se pueden utilizar todos los datos disponibles para realizar la predicción) ya que escapa del ámbito de la presente asignatura. Sin embargo, en otras asignaturas del máster (e.g., “Datos temporales y complejos”) se referirá con mayor detalle a este aspecto.

3. Exploración de datos

Una vez descargados los datos, es necesario realizar un extenso análisis descriptivo de los datos de entrenamiento, así como un proceso de inspección, limpieza y transformación de datos con el objetivo de resaltar información útil para la fase de modelado. Este análisis nos permitirá controlar la presencia de valores faltantes o la presencia de posibles errores en la fase de introducción de los datos (valores fuera de rango, inconsistencias, ...). También nos proporcionará una idea inicial de la forma que tienen los datos (distribución, parámetros de dispersión, ...), así como las relaciones entre los distintos atributos.

La función principal de esta exploración de los datos es que los alumnos utilicen las distintas gráficas, coeficientes y estadísticas utilizadas como base de cualquier decisión tomada para la generación de los consiguientes modelos de clasificación. Como posibles puntos de partida, el alumno puede analizar tanto los tipos de lanzamiento como su precisión en base a localizaciones (ver Figuras 6 y 7), temporadas, oponentes, tiempo, etc.

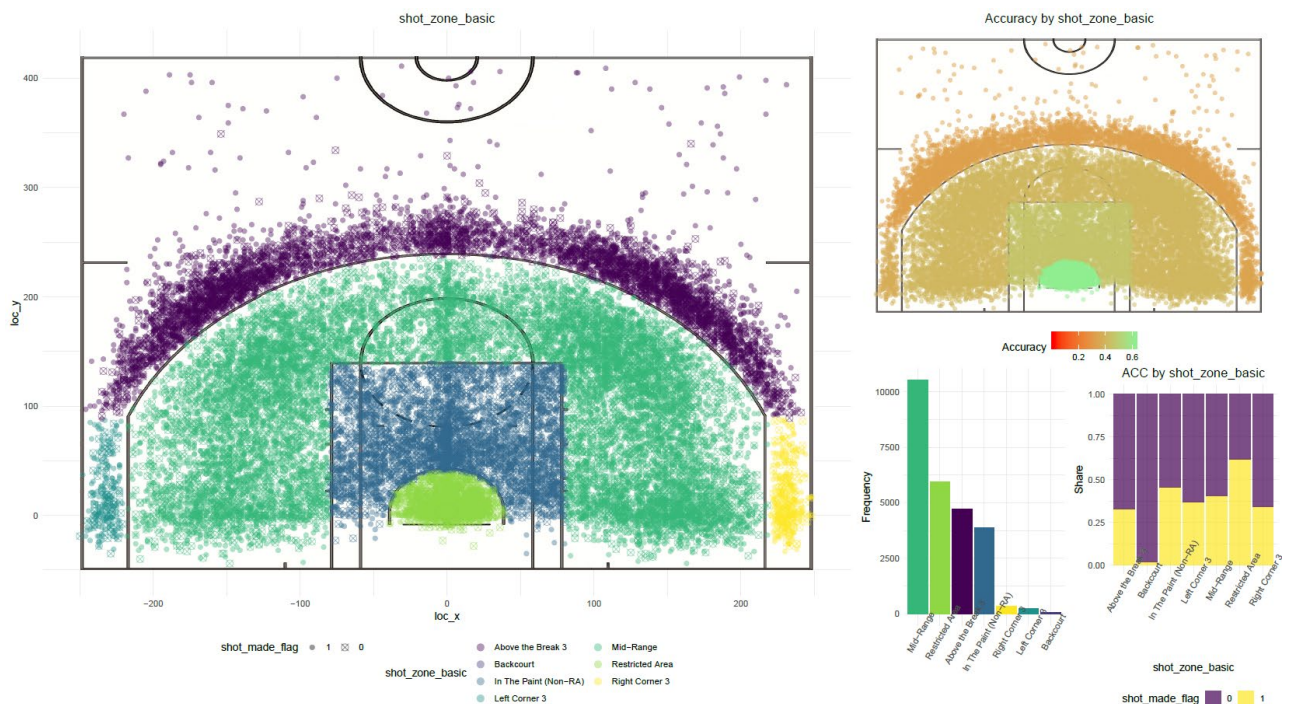


Figura 6. Ejemplo de análisis visual de la localización de los lanzamientos (variables *loc_x* y *loc_y*) por zonas (variable “*shot_zone_basic*”) y su precisión con respecto a la clase (“*shot_made_flag*”).

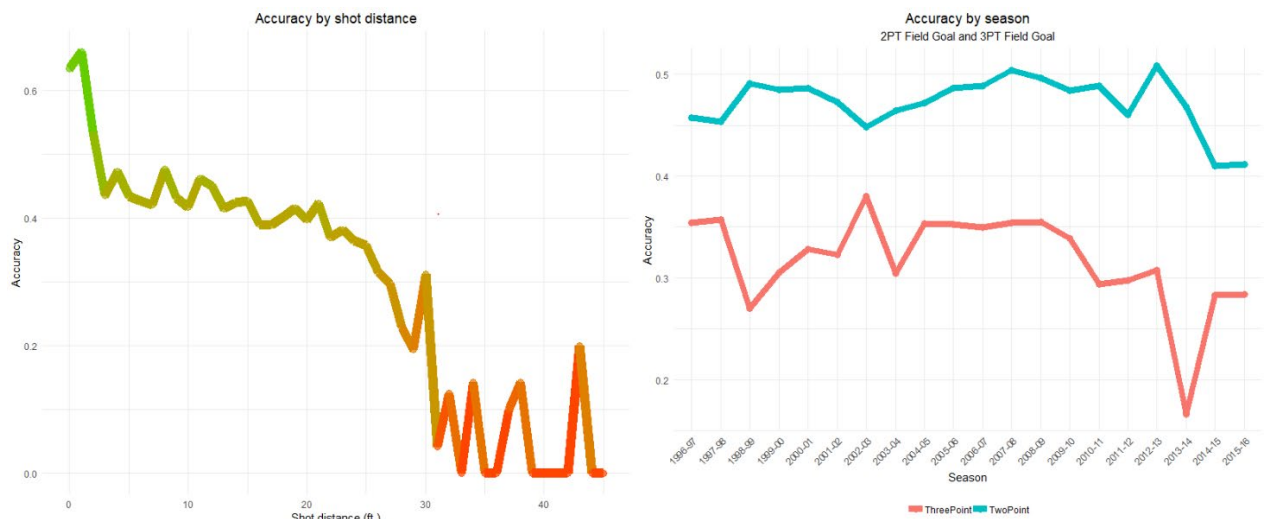


Figura 7. Ejemplo de análisis visual de la precisión de los por distancia (atributo "Shot distance") y temporada (atributo "Season")

TIPS

- Recuerda que estamos trabajando sobre el conjunto de datos de entrenamiento (train) y que todas aquellas operaciones de limpieza, transformación, creación de nuevos atributos, etc. que realicemos debemos también aplicarlas sobre el conjunto de datos de evaluación (test).
- Analiza la existencia de posibles valores anómalos, extremos o inusuales.
- Comprueba si en el conjunto de datos existen valores faltantes. Muchas técnicas NO pueden procesar observaciones con valores faltantes. Decide qué hacer con los valores faltantes en las variables numéricas (eliminación, transformación, sustitución por la moda/media/max/min, imputación...).
- Explora correlaciones entre atributos numéricos y la variable a predecir.
- Puedes transformar las variables categóricas en numéricas (e.g., "one-hot-encoding" para generar variables con 2 niveles), o numéricas en categóricas (e.g., binarización), aplicando distintas aproximaciones
- Para obtener un mejor ajuste de los datos, es necesario realizar un análisis más detallado de cada variable categórica con especial cuidado de aquellas variables que son ordinales, fechas, etc.
- Puede ser de utilidad la creación de nuevos atributos a partir de atributos ya existentes que permitan mejorar la descripción de los datos, así como reducir su dimensionalidad.
- También puede ser interesante crear nuevos atributos basados en la interacción de variables altamente correlacionadas
- ...

4. Predicción y Evaluación

Una vez hemos realizado nuestro análisis sobre los datos, **incluyendo la limpieza, transformación y generación de nuevas variables interesantes para nuestro estudio**, pasamos a la fase del modelado. Como ejemplo ilustrativo, podemos utilizar una técnica “sencilla” como un análisis discriminante lineal (o LDA). De forma resumida, un LDA es un método de clasificación supervisado en el que dos o más grupos son conocidos a priori y nuevas observaciones se clasifican en uno de ellos en función de sus características. Haciendo uso del teorema de Bayes, LDA estima la probabilidad de que una observación, dado un determinado valor de los predictores, pertenezca a cada una de las clases de la variable cualitativa, $P(Y=k|X=x)$. Finalmente se asigna la observación a la clase k para la que la probabilidad predicha es mayor. Es un método alternativo a la regresión logística que, en casos donde nos encontramos con un problema de clasificación con solo dos niveles, ambos métodos suelen obtener a resultados similares.

Para el entrenamiento de dicho modelo podemos utilizar la librería MASS en R:

```
> data <- as.data.frame(fread("data.csv", header = T, stringsAsFactors = T))
> train<-subset(data, !is.na(data$shot_made_flag))
> test<-subset(data, is.na(data$shot_made_flag))
> #. . . Limpieza y transformación de datos . . .
> library(MASS)
> dfl <- lda(shot_made_flag ~ ., train)
```

Si evaluamos los errores de clasificación en el conjunto de entrenamiento:

```
> pred.dfl <- predict(dfl, train)
> trainig_error <- mean(train$shot_made_flag != pred.dfl$class) * 100
> paste("Trainig_error =", trainig_error, "%")
[1] "Trainig_error = 38.3313227224968 %"
> confusionMatrix(as.factor(pred.dfl$class), as.factor(train$shot_made_flag))
```

Confusion Matrix and Statistics

	Reference	
Prediction	0	1
0	11978	7596
1	2254	3869

Accuracy : 0.6167
95% CI : (0.6107, 0.6226)
No Information Rate : 0.5538
P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.1876
McNemar's Test P-Value : < 2.2e-16

Sensitivity : 0.8416
Specificity : 0.3375
Pos Pred Value : 0.6119
Neg Pred Value : 0.6319
Prevalence : 0.5538
Detection Rate : 0.4661
Detection Prevalence : 0.7617
Balanced Accuracy : 0.5895

'Positive' Class : 0


```

LogLoss<-function(actual, predicted)
{
  predicted<-(pmax(predicted, 0.00001))
  predicted<-(pmin(predicted, 0.99999))
  result<- (-1/length(actual))*(sum((actual*log(predicted)+(1-actual)*log(1-
predicted))))
  return(result)
}
> LogLoss(train$shot_made_flag, pred.dfl$class)
[1] 4.413063

```

Con este modelo obtenemos un *Accuracy* igual a 0.61617 y un valor de *LogLoss*² (score utilizado en Kaggle como evaluación) igual a 4.413063, lo cual no parece demasiado bueno (aunque esperable dado el modelo utilizado). En este punto, podemos, entre otras muchas alternativas, intentar representar visualmente dónde se han producido los fallos de clasificación en base a la localización de los lanzamientos. Un ejemplo de esto es la Figura 8, donde podríamos aventurarnos a decir que una cantidad importante de errores de clasificación se producen en los lanzamientos más cercanos a la canasta (zona de 2 puntos), aunque esto debería corroborarse con un análisis más en profundidad.

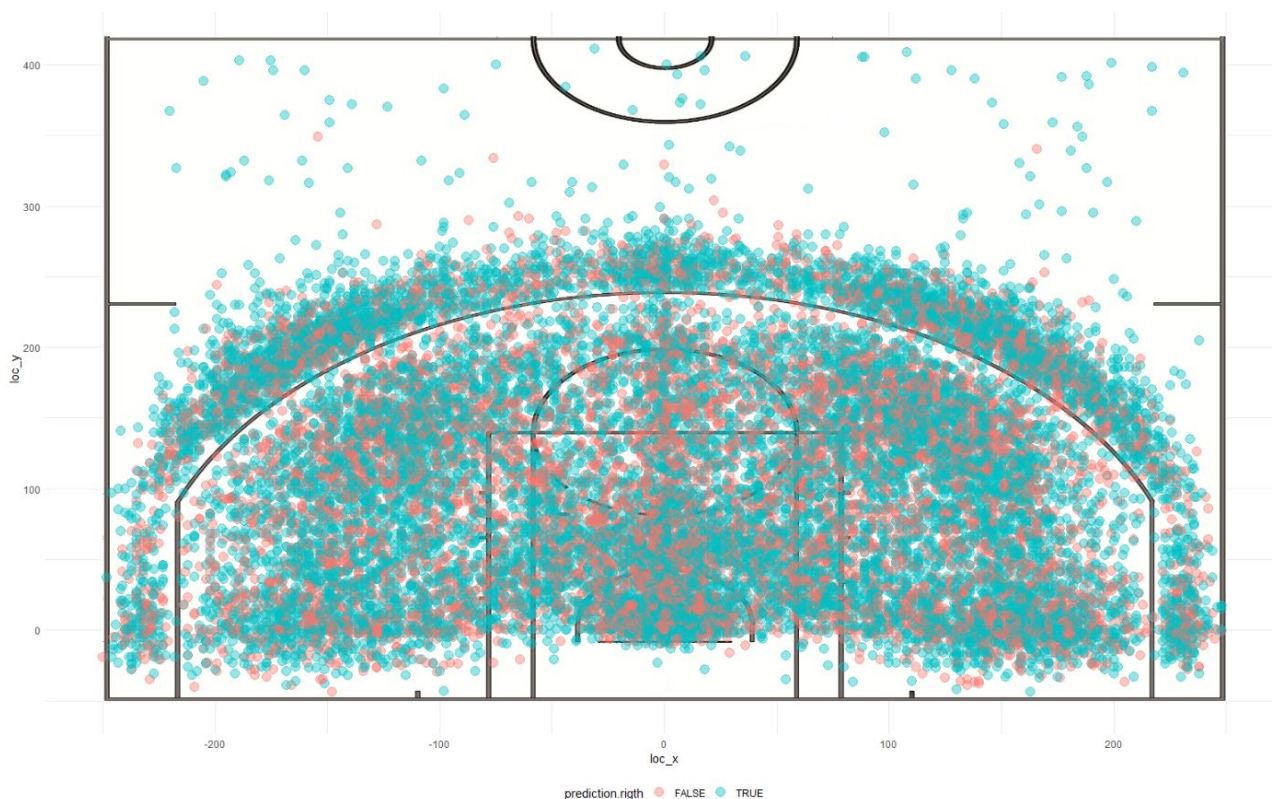


Figura 8. Ejemplo de análisis visual de los errores y aciertos de la clasificación en base a los atributos de distancia ("loc_x" y "loc_y").

² http://wiki.fast.ai/index.php/Log_Loss

Con el objetivo de obtener una evaluación de la predicción en Kaggle, es necesario subir los resultados obtenidos utilizando el conjunto de datos de evaluación (test) a la misma (apartado “Late submission”, Figura 10). Primero utilizamos el modelo para realizar la predicción. En R:

```
> pred <- predict(df1, test)
> samples <- data.frame(shot_id = test$shot_id, shot_made_flag = pred$class)
> write.csv(samples,file = "sumbission.csv", row.names = F)
```

Los envíos deben ser ficheros .csv, formados por dos columnas, “shot_id” y “shot_made_flag”(Figura 9).

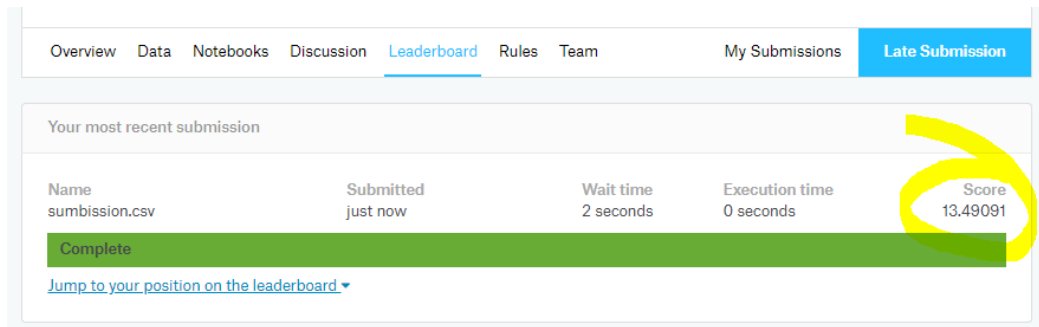
```
"shot_id","shot_made_flag"
1,"0"
8,"0"
17,"0"
20,"1"
33,"0"
34,"0"
35,"1"
```

Figura 7. Fichero csv para su evaluación en Kaggle.

The screenshot shows the Kaggle interface for the 'Kobe Bryant Shot Selection' competition. The 'Late Submission' tab is active, displaying a submission form. The form has two main steps: 'Step 1: Upload submission file' and 'Step 2: Describe submission'. Step 1 includes a file upload area with an upload icon and instructions on file format (CSV or zip/gz/rar/7z) and the number of predictions (5000 rows). Step 2 includes a text area for describing the submission. A 'Make Submission' button is at the bottom.

Figura 8. Pestaña “Late Submission”.

Para poder evaluar tu predicción debes volver a la competición en Kaggle -> “Late Submission”³ y subir el archivo .csv con la predicción y enviar. Tras unos segundos (si no hay errores) obtendrás el score obtenido (Figura 11). En este caso obtenemos un valor de LogLoss bastante peor que el obtenido con los datos de entrenamiento (una predicción aleatoria obtendría un valor de LogLoss cercano a 20). Esto nos indica que debemos trabajar más tanto en la fase de preprocesado de datos como en la de modelado.



Overview	Data	Notebooks	Discussion	Leaderboard	Rules	Team	My Submissions	Late Submission
Your most recent submission								
Name	Submitted	Wait time	Execution time	Score				
sumbission.csv	just now	2 seconds	0 seconds	13.49091				
Complete								
Jump to your position on the leaderboard								

Figura 9. Score obtenido en kaggle.

Ten en cuenta que sólo puedes subir un máximo de 5 ficheros de predicciones al día.

5. Objetivo de minería de datos

El objetivo de esta práctica es muy sencillo de explicar:

CONSTRUIR UN MODELO DE APRENDIZAJE QUE OBTENGA UNA ESTIMACIÓN LO MÁS PRECISA POSIBLE DEL ATRIBUTO “shot_made_flag” A PARTIR DEL RESTO.

6. Evaluación

El proyecto debe realizarse en R. Como entregable de este proyecto se realizará una memoria (ya sea un RMarkdown/notebook o un documento de texto) de **10 páginas como máximo** describiendo el proceso realizado en cada una de las fases de proyecto. Puede servir como guía (no exhaustiva):

- Introducción y descripción del problema.
- Comprensión y preprocesado de datos:
 - Exploración y visualización de datos
 - Limpieza (valores faltantes, outliers, ...)

³ Aunque la competición ya ha terminado, es posible realizar nuevos envíos obteniendo un resultado/posición (aunque esta no quedará reflejada en el “leaderboard” oficial de la competición).

- Procesado de variables numéricas (estudios de correlaciones, discretizaciones u otras transformaciones, ...)
- Procesado de variables categóricas (asociaciones/dependencias, numerizaciones (ordinales/nominales), ...)
- Construcción, formateo y estandarización de variables (creación de nuevas variables, normalizaciones, ...)
- Selección horizontal y/o vertical (selecciones de attrs basadas en filtros, modelos, correlaciones, pesos, ...)
- ...
- Modelado (técnicas de aprendizaje utilizada, construcción, comparativas, parámetros, ...)
- Evaluación (criterio de evaluación, resultados, comparativas, ...)

Finalmente, se deberá incluir también:

- Conclusiones (análisis de resultado, lecciones aprendidas tras la realización del proyecto, ...).
- Resultado en Kaggle (usuario y score obtenido mediante una captura de pantalla).

Se valorará muy positivamente la claridad de la memoria y la capacidad de síntesis. Aunque se hayan probado muchas cosas, los alumnos deben decidirse por un modelo únicamente, que es el que se evaluará en la plataforma Kaggle (generalmente el que hayan conseguido colocar más arriba, pero podría ser otro de su elección). **Es fundamental razonar y justificar todas las decisiones tomadas por el alumno en cada una de las fases.** Se deberá especificar claramente el usuario kaggle utilizado y el resultado obtenido para su comprobación (e.g., captura de pantalla).

Por otra parte, además de la memoria, **el profesor evaluará al alumno por medio de una entrevista (videoconferencia de 5 minutos) acerca del desarrollo del proyecto.** En la misma, el alumno describirá brevemente el desarrollo realizado y deberá responder acerca de cualquier particularidad del proyecto que el profesor se considere relevante. Una vez entregada la memoria, el alumno se pondrá en contacto con el profesor para concertar una cita (se proporcionarán más instrucciones en el foro de la asignatura).

Se deberán respetar todas las normas de kaggle y no usar otros usuarios adicionales al de la práctica para conseguir sobrepasar el límite de “submissions” por día que permite la plataforma.

La comunidad de usuarios de Kaggle proporciona una gran ayuda para la resolución de los distintos proyectos por medio de sus foros y “kernels” (scripts con código en R/Python/Matlab/...). Se recomienda al alumno la utilización de los mismos tanto para comenzar con el proyecto como para resolver dudas o coger ideas. **Cualquier pieza de código utilizada que provenga de cualquiera de estas fuentes debe ser citada en la evaluación.**

PARA LA NOTA FINAL SE VALORARÁ:

- **IMPORTANTE:** Que la predicción del modelo haya sido evaluada en kaggle. Cuanto menor sea el error cuadrático medio, mejor será la valoración por este concepto.
- **MUY IMPORTANTE:** Que se haya trabajado en el **preprocesado** (análisis, limpieza y transformación) de los datos y su comprensión.
- **IMPORTANTE:** Que se haya trabajado en el **modelado** intentando mejorar el resultado.
- **IMPORTANTE:** Que se haya realizado una **validación** correcta.
- **FUNDAMENTAL:** Que se responda adecuadamente al profesor y que de las respuestas no haya dudas de que el alumno ha realizado la práctica y que sabe manejar mínimamente las herramientas que hayas utilizado.