UNIVERSIDAD DE LA FRONTERA

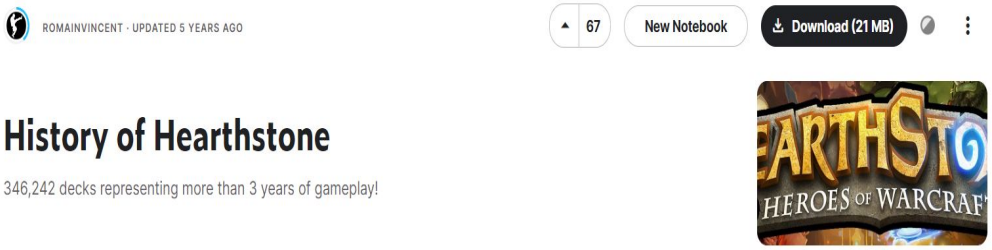
DEPARTAMENTO DE
CS. COMPUTACIÓN E INFORMÁTICA

# Hito 3 - Ing. de Datos
# Historia de Hearthstone

Rodrigo Valenzuela
Armin Rodríguez
Pablo Nahuelpán
Nicolás Hidalgo
Noviembre, 2022

# Motivación y Dataset

- El Dataset escogido inicialmente contaba con 346.232 mazos, estos siendo desde la beta hasta finales de 2017.
  Sus atributos principales eran las cartas que se hallaban en cada mazo, el set, los arquetipos, entre otros.




head ( ) del dataset

| date | deck_archetype | deck_class | deck_format | deck_set | deck_type | card_0 | card_1 | card_2 | card_3 |
|---|---|---|---|---|---|---|---|---|---|
| 2016-04-26 | Mill Rogue | Rogue | S | Old Gods | Ranked Deck | 180 | 180 | 196 | 196 |
| 2016-04-26 | N'Zoth Hunter | Hunter | W | Old Gods | Ranked Deck | 296 | 296 | 437 | 437 |
| 2016-04-26 | Unknown | Druid | S | Old Gods | None | 64 | 64 | 95 | 137 |
| 2016-04-26 | C'Thun Priest | Priest | S | Old Gods | Ranked Deck | 272 | 272 | 613 | 613 |
| 2016-04-26 | Unknown | Mage | S | Old Gods | None | 138 | 138 | 172 | 172 |

# Limpieza de datos

**Variables no relevantes**

- craft_cost
- title
- user
- rating
- deck_id

**Variables con datos faltantes, erróneos o nulos**

- deck_archetype

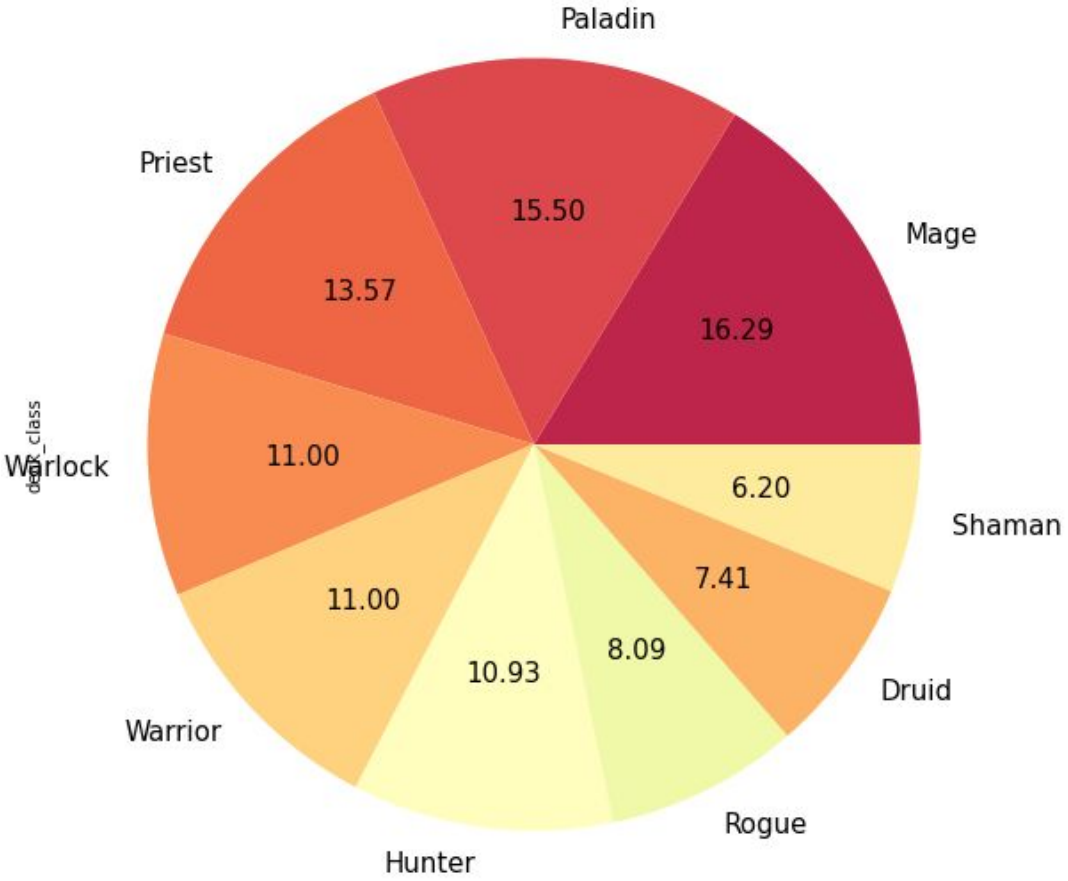
```
filter_deck = filter_deck[filter_deck['date'] > '2014-03-11']
print ('Primero Mazo :', min(filter_deck['date']))
print ('Ultimo Mazo :', max(filter_deck['date']))
```

```
Primero Mazo : 2014-03-12 00:00:00
Ultimo Mazo : 2017-12-03 00:00:00
```

- **Se realizó un filtrado de datos por set “Old God”, ya que sería lo principal a tener en cuenta en nuestra exploración de datos (EDA).**

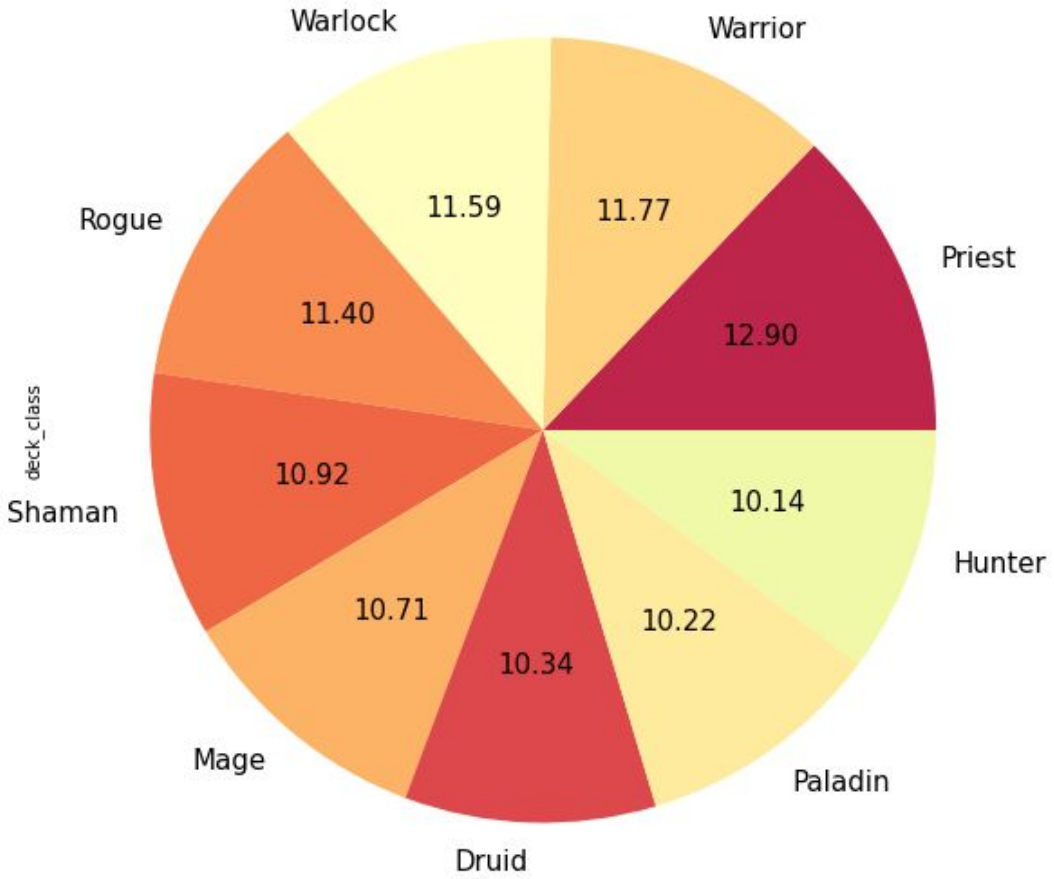
**Dataset Original: 346.232**
**Dataset post limpieza: 30.087**

# Exploración de datos




*Razas populares en formato Salvaje*

# Exploración de datos




*Razas populares en formato Estándar.*

# Preguntas y problemas

- Problema de clasificación binaria basado en la variable "deck_format": consiste en una variable que es categórica que se comporta de manera binaria correspondiente a un atributo que hace referencia al formato de un mazo, y en este problema se debe encontrar la forma de poder clasificar el formato de los nuevos mazos que serán registrados en la plataforma.

- Problema basado en encontrar asociaciones entre variables categóricas de un mazo. Dado que se registraron mazos con sus respectivos atributos, en este problema se debe encontrar cuáles son los atributos de un mazo que se asocian entre sí para obtener información relevante, ya que dentro de la plataforma se quiere implementar una sección de recomendación a través de la información que se obtenga de las asociaciones de atributos.

# Propuesta experimental

- Para enfrentar el problema 1, vamos a clasificar con 2 modelos de predicción de tipo categórico, estos son árbol de decisión y bosques aleatorios con el fin de identificar qué modelo es mejor para poder clasificar el formato de un nuevo mazo, y los resultados que nos entreguen los dos modelos se evaluaran respecto a la métrica de "Recall", debido a que el valor para este problema proviene de ser capaces de detectar decks Wild, los cuales son considerados como el TP de esta situación.

- Para enfrentar el problema 2, vamos a utilizar reglas de asociación de los atributos de un mazo para encontrar reglas de asociación utilizando un modelo con el que será entrenado con el algoritmo Apriori y esto será medidos por la métrica de "support" y la razón de elegir esta medida es porque nos entrega un valor de la cual podemos saber si se puede confiar en las transacciones para obtener la información que necesitamos dependiendo del valor que nos entregue.

# Resultados Experimento 1

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| S | 0.99 | 0.99 | 0.99 | 6884 |
| W | 0.93 | 0.93 | 0.93 | 638 |
| accuracy | | | 0.99 | 7522 |
| macro avg | 0.96 | 0.96 | 0.96 | 7522 |
| weighted avg | 0.99 | 0.99 | 0.99 | 7522 |

Matriz de confusión:
[[6841 43]
[ 43 595]]

Árbol de decisión

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| S | 0.99 | 1.00 | 1.00 | 6828 |
| W | 1.00 | 0.93 | 0.96 | 694 |
| accuracy | | | 0.99 | 7522 |
| macro avg | 1.00 | 0.96 | 0.98 | 7522 |
| weighted avg | 0.99 | 0.99 | 0.99 | 7522 |

Matriz de confusión:
[[6828 0]
[ 49 645]]

Random forest

# Nueva dirección

- Enfoque basado en agrupación con Kmodes, variación para datos categóricos del algoritmo Kmeans.
- Se convierten clases de “deck_format” desde “S” y “W” a binario para la utilización de este método.
- Se remueve atributo “date” debido a que no es un atributo categórico.

# Experimento 1.2

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.91 | 0.86 | 0.88 | 13757 |
| 1 | 0.08 | 0.13 | 0.10 | 1287 |
| accuracy | | | 0.80 | 15044 |
| macro avg | 0.50 | 0.49 | 0.49 | 15044 |
| weighted avg | 0.84 | 0.80 | 0.82 | 15044 |

Matriz de confusión:
[[11792 1965]
[ 1118 169]]
Costo de prediccion: 417235.0

Resultado de clasificación Kmodes

| | |
|---|---|
| 0 | 27445 |
| 1 | 2642 |

Distribución de clases

# Experimento 1.2

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.60 | 0.78 | 0.68 | 6932 |
| 1 | 0.67 | 0.47 | 0.55 | 6792 |
| accuracy | | | 0.62 | 13724 |
| macro avg | 0.64 | 0.62 | 0.61 | 13724 |
| weighted avg | 0.64 | 0.62 | 0.61 | 13724 |

Matriz de confusión:
[[5399 1533]
[3629 3163]]
Costo de prediccion: 384617.0

Mejor resultado Kmodes, luego de sampling.

# Resultados Experimento 2

- Limpieza de atributos, que afectan el rendimiento de la regla de asociación.

```
%%R
decks_oldsgods <- decks_oldsgods[, -c(6:36)]
decks_oldsgods$date <- decks_oldsgods$deck_set <- NULL
head(decks_oldsgods)
```

```
     deck_archetype deck_class deck_format   deck_type
1407     Mill Rogue      Rogue           S Ranked Deck
1497  N'Zoth Hunter     Hunter           W Ranked Deck
1499  C'Thun Priest     Priest           S Ranked Deck
1623        Zoolock    Warlock           S Ranked Deck
1628        Zoolock    Warlock           W Ranked Deck
1632   Aggro Shaman     Shaman           S Ranked Deck
```

# Resultados Experimento 2

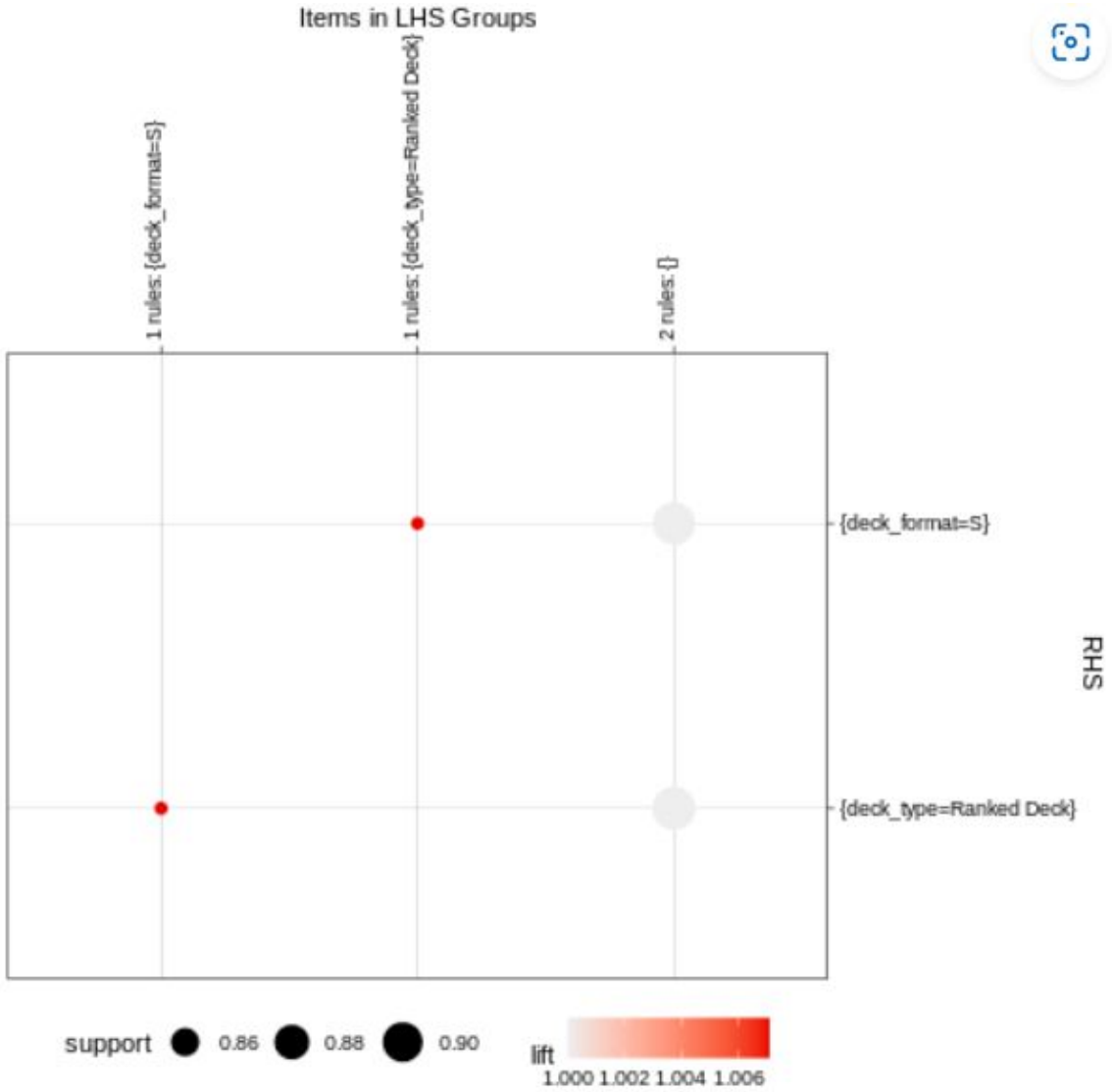
```
    lhs                          rhs                          support   confidence
[1] {deck_format=S}          => {deck_type=Ranked Deck} 0.8411274 0.9220987
[2] {deck_type=Ranked Deck}  => {deck_format=S}         0.8411274 0.9186511
[3] {}                       => {deck_format=S}         0.9121880 0.9121880
[4] {}                       => {deck_type=Ranked Deck} 0.9156114 0.9156114
    coverage  lift     count
[1] 0.9121880 1.007085 25307
[2] 0.9156114 1.007085 25307
[3] 1.0000000 1.000000 27445
[4] 1.0000000 1.000000 27548
```

- Se usó support para medir el nivel de confianza de la regla.

**Itemsets**

```
    items                                 support   count
[1] {deck_type=Ranked Deck}               0.9156114 27548
[2] {deck_format=S}                       0.9121880 27445
[3] {deck_format=S, deck_type=Ranked Deck} 0.8411274 25307
```

# Resultados Experimento 2




- Existen asociaciones entre las variables "deck_format" y "deck_type" con un "support" de 0.8411274 lo que es un buen indicador para confiar en estas dos asociaciones.

# Futuras direcciones

UNIVERSIDAD DE LA FRONTERA

DEPARTAMENTO DE
CS. COMPUTACIÓN E INFORMÁTICA

# Hito 3 - Ing. de Datos
# Historia de Hearthstone

Rodrigo Valenzuela
Armin Rodríguez
Pablo Nahuelpán
Nicolás Hidalgo
Noviembre, 2022