Diseño del DAaaS

### **Definición la estrategia del DAaaS**

*El presente proyecto tiene por objetivo el implementar un sistema de predicción de las loterías en la web (bonoloto) apoyándonos en tres estrategias:*

***La estadística****:* Con la que conseguiremos métodos, procedimientos y fórmulas que nos permitirá analizarla y extraer de ella conclusiones relevantes. Aplicando todo el conjunto de parámetros estadísticos de todos los sorteos hasta la fecha de hoy podríamos obtener otro conjunto de apuestas mucho mas reducidas con las mismas garantías de acceso al premio mayor. El global de las todas las apuestas suman un total de 13.893.816 posibles combinaciones. SI tomamos como base 200 o 300 apuestas de forma aleatoria y extraemos sus parámetros estadísticos (media, desviación, sumas, etc.) para ponerlos como filtro, estas 13.893.816 apuestas, pueden ser reducidas a casi la mitad del número de combinaciones posibles. Podemos trabajar sobre este y no sobre las 13.893.816 apuestas.

**La relación entre los números ganadores y los distintos parámetros estadísticos**: Los distintos parámetros estadísticos nos sirven para sintetizar o resumir la información de un sorteo ganador, pero no nos dice nada acerca de cómo se relacionen los números ganadores con esos parámetros estadísticos. innovaremos aplicando distintos métodos para relacionar números ganadores y parámetros estadísticos transformando los datos a una base de datos de grafos.

**Ley del tercio: La ley del tercio constata que, para un ciclo de juego, un tercio de los números no aparecen en beneficio de los dos tercios restantes, de los cuales otro tercio aparecerá 1 vez, y el resto 2 o más veces. Utilizaremos Ciclos cortos (los últimos 7 sorteos) para definir los algoritmos. Si observamos la frecuencia con la que han aparecido los números en todos los sorteos observamos que todos ellos tienen un porcentaje de salida en los sorteos muy próximos entre sí. Sin embargo, cuando se estudian ciclos cortos de sorteos (7 sorteos, 16 sorteos,21 sorteos) esta tendencia parece romperse y empiezan a aparecer muchos patrones que podemos aprovechar para nuestras estrategias. Por ejemplo, en un ciclo de 7 sorteos hay algunos números que muestran una frecuencia de salida muy por encima de otros (salen 2, 3 e incluso 4 veces en ese ciclo)**

### 

ANEXO 🡪 Csv

### **Arquitectura DAaaS**

1º DESARROLLO DE LOS ALGORITMOS.

* Herramienta ETL 🡪 Talend Studio
* Bucket para el almacenaje del csv y grabación de los resultados
* Instancias y soluciones del Marketstore para el tratamiento y análisis de los datos: MySQL, Elasticsearch, Neo4J Aura.
* Clúster de Hadoop 🡪 DATAPROC (Hive, Kibana)
* Herramientas de Machine Learning en la Nube (Vertex AI)

2º DISEÑO Y MONTAJE DE LA WEB.

* Servicio de Hosting y Domino WEB con soporte MySQL en:
* <https://www.hostinger.es> 🡪 [www.lavalotto.com](http://www.lavalotto.com)
* Servicios en la Nube: Cloud Function
* Base de Datos MySQL para clientes y Base de datos MySQL para los sorteos
* Crawler para la extracción del último sorteo (BeatifoulSoap):
  + [www.loteriasyapuestas](http://www.loteriasyapuestas).

ANEXO 🡪 (Crawler)

### **DAaaS Modelo de Operaciones.**

El Modelo de operaciones esta dividido en dos partes. Una (No automatizada que consiste en Trasformar los datos, analizarlos, extraer conclusiones y hacer algoritmos predictivos con las diversas herramientas de Machine Learning. Una Segunda parte que consiste en montar una web para ofrecer los servicios de apuestas optimizadas.

PARTE 1

1. Hacer un proceso de ETL sobre nuestra fuente de datos CSV. Extraer los sorteos y los parámetros estadísticos y transportarlos a una Base de Datos relacional (MySQL), No relacional (Elasticsearch) y Base de datos de Grafos (Neo4j). Utilizamos Un bucket de Google Storage para este proyecto.
2. Montamos un DATAPROC para aprovechar las herramientas de procesamientos **por lotes, búsquedas, transmisiones y aprendizaje automático.**
3. Analizamos los datos y el análisis y los resultados de esos análisis nos servirán de base para diseñar los algoritmos
4. Creamos Una CLOUD function (Con activación por URL) por cada algoritmo que creamos.

PARTE 2

1. **Diseño de la interfaz gráfica y de la web con Django**. La interfaz grafica desarrolla un diseño en forma de mando de un selector de lavadora, de forma que cada selección corresponde a un algoritmo que se ejecuta en una CLOUD FUNCTION. Algunos ejemplos:
   1. **Programa “Lavado Mixto”**: contiene un algoritmo que evalúa el ciclo de los 7 últimos sorteos y proporciona un 50% de los números más caliente y otro 50% de los números que no han salido en ese ciclo.
   2. **Programa “Aclarado y Centrifugado”:** Sobre una reducción de apuestas al 5 (147.111 apuestas, garantizando el premio de 5 aciertos) aplicamos los algoritmos de optimización
2. Contratamos los servicios de Hosting y dominio Web con el proveedor

<https://www.hostinger.es>

precio: 3,99 Mes.

1. Montar La Base de Datos de clientes y exportamos la Base de Datos de apuestas de MySQL que hemos utilizado para nuestros análisis en la primera parte a nuestro servidor web.
2. Montamos el CRAWLER en una CLOUD FUNCTION con un CLOUD Schedule para que se actualice la Base de Datos de los Sorteos todos los días sobre las 00:00. Utilizamos un trigger en la Base de Datos MySQL como disparador en la inserción de estos datos de forma que se calcule el resto de condiciones estadísticas para ese sorteo.
3. Puesta en Marcha de la web.
4. El usuario podrá seleccionar diversas opciones como incluir sus números favorito, seleccionar el número de apuestas, etc. Cada vez que un usuario realice una selección, se le devuelve el resultado con una CLOUD FUCNTION .Como el precio de las Cloud function esta establecido es fácil establecer un precio por apuesta.

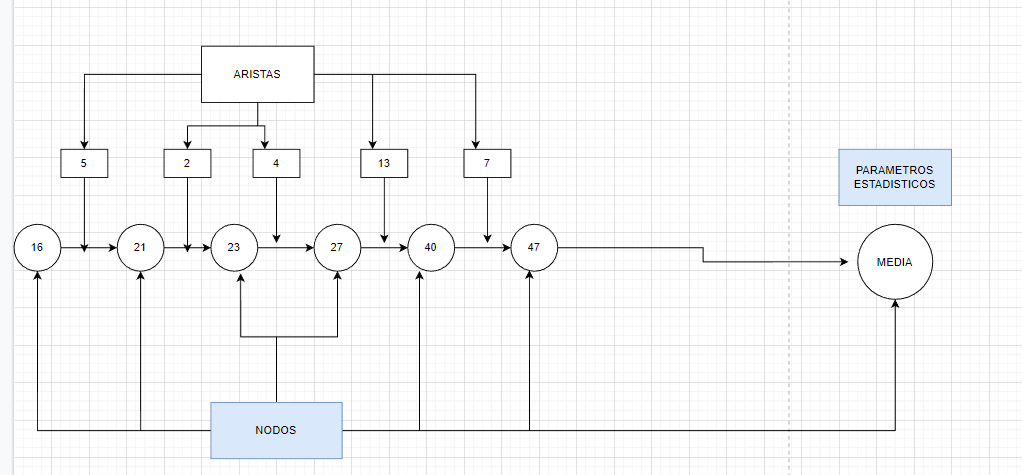
### **Desarrollo de la plataforma DAaaS.**

*Básicamente, un paso muy importante es el proceso de ETL de los datos del csv, el modelado de la base de datos, poque serán la base para obtener nuestros algoritmos.*

*Incluimos elasctic search como motor de búsquedas para nuestros análisis ( aunque básicamente esta orientado a texto) porque muchos de nuestros campos o columnas en nuestra Base de datos son tratados como String. Por ejemplo, sobre las figuras de Pares impares (“3+3“) o las figuras de terminaciones (“21111”🡪 dos números con la misma terminación y el resto diferente). Incluir elasticSearch nos proporcionara valiosa información para nuestros análisis.*

*Sin embargo, es la trasformación desde los datos de una base relacional a una base de datos de grafos la verdadera novedad porque vamos a poder establecer patrones que correlacionan los números ganadores de un sorteo con los datos o parámetros estadísticos.*

*El procedimiento será transformar cada sorteo en un grafo, tomemos por ejemplo el ultimo sorteo ganador 🡪16-21-23-27-40-47 (8-10-2022). Este sorteo puede ser transformado a grafo, siendo los nodos los números ganadores y las aristas la diferencia entre el primer número y el segundo:*

**

*Podríamos hacer una selección por ejemplo de la Media aritmética que mas ha salido y ver que relaciones existe entre los números ganadores y ese dato estadístico. Se observa que en este sorteo las aristas son únicas (no se repiten) por lo que forman una figura de DISTANCIAS ->11111.Tambien que la figura de terminaciones es 2111. Que existe un salto de una unidad de un numero hacia el otro (21-23). Que existen predominancia de los impares sobre los pares, etc. Si hiciéramos un select de todos los sorteos con la misma media podríamos obtener bastante información de cómo se combinan los números ganadores y que números presentan patrones interesantes.*

*De igual forma podrían establecerse diferentes maneras de relacionar los nodos entre sí: resta, la suma, números reversibles (11,22,33,44), primos, etc. Incluso podría acotarse aún más el análisis, extrayendo las conclusiones si seleccionamos la media que mas sale y seleccionando los días por separado (lunes-martes-miércoles…). Quizás es posible descubrir que por ejemplo los martes siempre ocurre un mismo patrón.*

*Por todo ello la parte de más trabajo y arduo será la de análisis y el diseño de los algoritmos.*

*La segunda parte es más sencilla, su implementación no requiere de tanta complejidad.*

Link a Diagrama:

