DataKnow

Prueba Técnica de Conocimiento

Perfil Analítico – Ingeniero de Datos – Científico de Datos

Muchas gracias por tu interés en participar en la convocatoria para pertenecer a la familia DataKnow. **AI Team DataKnower**

Estamos buscando personas comprometidas, que se destaquen por realizar un trabajo de calidad, con buena actitud de servicio, compromiso y mucha responsabilidad con sus actividades día a día, siempre dando prioridad a las necesidades del cliente. Además, también estamos buscando personas con ganas de adquirir sólidos conocimientos en técnicas de modelación, con habilidades en estadística, matemáticas, bases de datos, big data, nube (Azure, AWS, GCP), con fuertes habilidades en programación en leguajes como: R, Python, SAS, PL/SQL, Scala, Hadoop, entre otros.

El propósito de esta prueba es medir sus capacidades para manipular datos de diferentes industrias, realizar supuestos, filtrar y utilizar información relevante, concluir y comunicar adecuadamente los resultados de los modelos. Pruebe usar cualquier herramienta de programación.

1. CARGA DE INFORMACIÓN

Cargar un data set, realizar el cargue y depuración del archivo OFEI1204.txt.

Se debe entregar una tabla con las columnas:

Agente

Planta

Hora_1

Hora_2

Hora_3

...

Hora_24

Solamente procesar los registros Tipo D.

Enviar junto con la tabla resultante el código utilizado.

Explicar el paso a paso en un archivo de texto (.doc o .pdf).

2. MANIPULACIÓN DE DATOS

- Cargar un data set, del archivo Excel Master Data, únicamente las siguientes columnas:
- Nombre visible Agente
- AGENTE (OFEI)
- CENTRAL (dDEC, dSEGDES, dPRU…)
- Tipo de central (Hidro, Termo, Filo, Menor)
- Seleccionar los registros que pertenecen al agente EMGESA ó EMGESA S.A. y adicionalmente que el Tipo de Central sea 'H' o 'T'. Cargar el archivo dDEC1204.TXT que viene por Central.
- Realizar el merge de los dos data sets por Central.
- Calcular la suma horizontal de todas las horas para cada planta.
- Seleccionar solamente los registros de las plantas cuya suma horizontal sea mayor que cero.
- Los resultados deben ser entregados en un dataset.
- Enviar junto con la tabla resultante el código utilizado.
- Explicar el paso a paso en un archivo de texto (.doc o .pdf).

3. PRUEBA DE SQL

Utiliza cualquier dialecto de SQL de tu elección para abordar estos desafíos, de preferencia genera los datos si lo ves necesario para simular y emplear las soluciones de diseño, la idea es explicar tu solución de tal forma que técnicamente el equipo pueda ser capaz de entender y visualizar usa las herramientas que desees además de hacer los scripts de creación dependiendo de cada parte de la prueba.

- **Parte 1** Un interesado nos solicita prepararnos para una nueva fuente de datos dentro de nuestro entorno de almacenamiento de datos. La tabla recogerá información meteorológica de forma horaria para diferentes regiones. Las dimensiones y métricas de la tabla deben crearse con los tipos de datos apropiados. La tabla debe diseñarse de manera que se pueda identificar de forma única cada registro dentro de ella. Las siguientes columnas deben estar presentes en la definición de la tabla:
  - Localidad (Poblados en Medellín, Envigado, Sabaneta, etc.)
  - País (Colombia)
  - Temperatura (Grados Celsius)
  - Fecha y hora del registro (horario)
  - Cobertura de nubes (Mínima, Parcial, Total)

- Índice U/V
- Presión atmosférica
- Velocidad del viento (Nudos)

- **Parte 2** La tabla definida en la Parte 1 se implementa y comienza a recopilar datos. La tabla se vuelve considerablemente grande, con millones de registros. Proporciona tres maneras en que la tabla actual puede mejorarse para manejar un conjunto de datos más grande y mantener una óptima legibilidad de los datos.
- **Parte 3** El mismo interesado llega con nuevos requerimientos. Además de la tabla ya existente, se requiere una nueva tabla completamente separada que recopile la misma información, pero registre la temperatura en Fahrenheit (en lugar de Grados Celsius). Además, la nueva tabla contendrá los registros de temperatura distribuidos por día, en lugar de por hora. La nueva tabla debe contener todos los datos ya recopilados de la tabla definida en la Parte 1.
- **Parte 4** Se recibe un nuevo requerimiento por parte del interesado. Ambas tablas definidas en la Parte 1 y la Parte 3 deben ahora capturar la diferencia de temperatura (delta) entre un registro y el anterior. En el caso horario, la nueva métrica contendrá la diferencia entre el momento actual y una hora antes. Para el caso diario, la nueva métrica contendrá la diferencia entre el momento actual y el día anterior. La nueva columna debe ser completada retroactivamente para todas las temperaturas ya existentes.

4. PRUEBA DE AWS

Construya una solución completa en la nube de AWS que usando todas las tablas de la fuente de datos de muestra de

Redshift: https://docs.aws.amazon.com/redshift/latest/gsg/samples/tickitdb.zip

Puede encontrar más información sobre los tipos y columnas de estos datos en:

Step 6: Load sample data from Amazon S3 - Amazon Redshift

Realice las siguientes tareas y guarde el código y evidencia de ejecución de las mismas:

1. Configure los roles específicos de los recursos AWS necesarios para todo el ejercicio
2. Configure un Clúster de pruebas de Redshift con los requerimientos mínimos necesarios

3. Copie la información a Redshift usando algún editor de queries como SQL Workbench/J - Home (sql-workbench.eu)
4. Responda las siguientes preguntas usando comandos de consultas SQL:
   a. ¿Cuántos Usuarios gustan del Jazz?
   b. ¿Cuántos Usuarios gustan de la ópera y del rock al mismo tiempo?
   c. ¿Cuál es el promedio, moda y mediana del total de Ventas?
   d. ¿Cuál el promedio de ventas de usuarios que gustan del rock, pero NO del Jazz?
5. En una nueva tabla junte la información (Nombre de usuario, Apellido de usuario, Correo de usuario, Nombre del evento, lugar del evento, Fecha del evento, Cantidad y Total vendidos) y expórtela usando Redshift a un bucket predefinido de S3.
6. (Opcional) Sobre la data exportada en el punto 5, y usando cualquier información adicional que desee, cree una sesión de SageMaker y realice un modelo de Forecast con cualquier técnica de su preferencia, para pronosticar las ventas para los siguientes 7 días desde el final del histórico de datos. Tenga en cuenta que la fecha de la venta se encuentra en la variable saletime y que esta está mostrada a una granularidad de factura individual.

Opcional: se puede resolver sobre tecnología Azure.

5. PRUEBA DE AZURE

Construya una solución completa en la nube de Azure que usando todas la base, de pruebas adventure Works, permita crear una etl, para la realización de un trabajo de reporteria dentro de la organización.

- desplegar base de datos en sql, con la base de pruebas adventure works
- realizar un pipeline con Azure Datafactory, utilizando data flow, para realizar la carga de una base de datos, crear 5 indicadores.
- realizar una etl, que poble un datalake.

6. PRUEBA DE MODELACIÓN ANALÍTICA

El archivo train.csv contiene información sobre muchas transacciones con tarjetas de crédito y débito por diferentes canales. Para cada transacción se tiene el valor monetario de la misma y otras variables (ver diccionario_variables.xlsx). De particular importancia es la variable FRAUDE en donde aparece 1 si la transacción constituyó un fraude o 0 si fue una transacción legítima. Su misión es desarrollar un

modelo que permita, a partir de los datos en este archivo predecir cuál será el valor de la variable FRAUDE para una transacción cualquiera. El archivo test.csv contiene exactamente las mismas columnas de train.csv, la columna FRAUDE la dejamos en blanco.

1. Cargue el archivo train.csv y Construya un modelo que capaz de realizar predicciones de FRAUDE.
2. Enviar un archivo test_evaluado.csv con todas las columnas en el mismo orden que se encuentran en test.csv y adicionalmente la columna FRAUDE poblada con el valor predicho por su modelo. Cualquier valor real (es decir, fraccionario) entre 0 y 1 será admisible aquí, donde 1 debe corresponder a FRAUDE y 0 a transacción legítima.
3. *opcional, realizar montaje de este sobre servicios azure o AWS, realizar puesta en productivo batch o/y servicio.

Nota: Muy importante enviar un archivo de texto (.doc o .pdf) donde se documente muy bien cada paso realizado, se muestren claramente los resultados y análisis respectivos, adicionalmente se deben enviar todos los códigos y comandos (con comentarios) utilizados para desarrollar esta prueba.

7. Arquitectura

En la empresa gaseosas SA están trabajando en una solución analítica que sea capaz de procesar miles de datos de las ventas donde se describen comportamiento de compra y análisis previos hechos por vendedores a clientes con gran volumen de compra, de forma rápida y confiable mediante el uso de tecnologías Big Data de analítica, para entrenar un modelo que sea capaz de identificar los patrones de estas ventas y compararlos en tiempo real con los patrones de datos capturados de manera streaming por dispositivos implantados puntos de venta, para controlar tempranamente y evitar el desabastecimiento.

Tu tarea es realizar un correcto diseño de la arquitectura para la solución analítica que podría soportar estos requerimientos. (Ilustra tu diseño y da una breve explicación de su funcionamiento), es importa definir el gobierno de datos y modelos de acuerdo a los perfiles

MUCHAS GRACIAS, MUCHOS EXITOS!!!!!!!!