Ejercicio 1

Crear un flujo de trabajo que debe realizar las siguientes operaciones:

- Leer el archivo data1.txt con la columna "ranking" como String y denominada "marks";
- Eliminar los comentarios iniciales de los datos leídos del archivo;
- Eliminar la columna "class"
- Escribir los datos finales en el archivo en formato CSV (por ejemplo, con el nombre "data1_new.csv") utilizando el carácter ";" como separador

Escribe una breve descripción de todos los nodos del flujo de trabajo.

Guarda y ejecuta el workflow. La ejecución debe realizarse sin errores (luces verdes para todos los nodos).

Ejercicio 2

Partiendo del fichero CSV resultado del ejercicio anterior. Ahora crea un workflow que haga lo siguiente:

- Leer el archivo CSV y cambiar el nombre de la columna “marks” a “ranking”
- Filtrar las filas con valor ‘average’ en la columna ‘comments’
- Excluir las columnas con datos del tipo Integer
- Escribir los datos finales en el archivo en modo "Append" y con tab como carácter de separación
- Cambia el nombre de todos los nodos cuando sea necesario. Guarda y ejecute el workflow.

Ejercicio 3

A partir del dataset del Titanic, se pide realizar un workflow que efectúe los siguientes pasos:

- Cargar el fichero CSV desde una ruta relativa al *workflow*, en su subcarpeta *data*.
- Convertir la columna *Survived* a *string*
- Convertir la columna *Sex* a tipo entero (*Category To Number*), sobrescribiendo la propia columna
- Eliminar la columna *Cabin*, porque tiene demasiados nulos
- Reemplazar las edades nulas por la media de edad y los *Embarked* nulos por el valor más frecuente de esa columna (puedes hacerlo todo junto con un solo *Missing Value*)
- Guarda el resultado procesado con un *CSV Writer* en un fichero en la misma carpeta que el original

- Paralelamente, desde el penúltimo nodo, filtra los supervivientes, y quédate con la columna *Pclass*. Convierte este dato a *string* y saca un gráfico de barras de cuántos supervivientes hubo por clase.

Ejercicio 4

Vamos a realizar un proyecto más completo, con más fases. Recopilaremos datos de diferentes fuentes, los limpiaremos y combinaremos para producir un resultado final. Usaremos el fichero del dataset para el ejercicio 4, en él veremos varios archivos que son los que nos interesan:

- Product Info: información de los productos en venta de una compañía (bebidas energéticas), se indican precios de venta y coste de fabricación en cada país donde se venden.
- Sales Rep: información sobre las diferentes regiones de venta, los productos que se venden en cada región y quién es el responsable o jefe de ventas de ese producto en esa región.
- Finalmente, un conjunto de archivos CSV con las ventas realizadas en diversos meses. Se indican en cada registro su ID, la región donde se vendió, el nombre de producto que se vendió, país y estado, cantidad, fecha y precio que se pagó.

Crea un workflow group donde añadiremos 3 workflows diferentes. El objetivo es conseguir en cada uno lo siguiente:

- UnificacionTransacciones:
    - Usa *List Files/Folders* y especifica la carpeta donde están los ficheros. En Filter options configurar archivos con extensión csv y que empiecen por Transactions.
    - Itera sobre esos ficheros usando Table Row to Variable Loop Start
    - Luego usaremos un CSV Reader conectado a la salida del nodo anterior mediante *Show Flow Variable Ports* No obstante, le diremos que lea la variable Path que proporciona el nodo anterior.
    - Finaliza con un Loop End y luego un CSV Writer. Pídele que sobrescriba el fichero si ya existía.

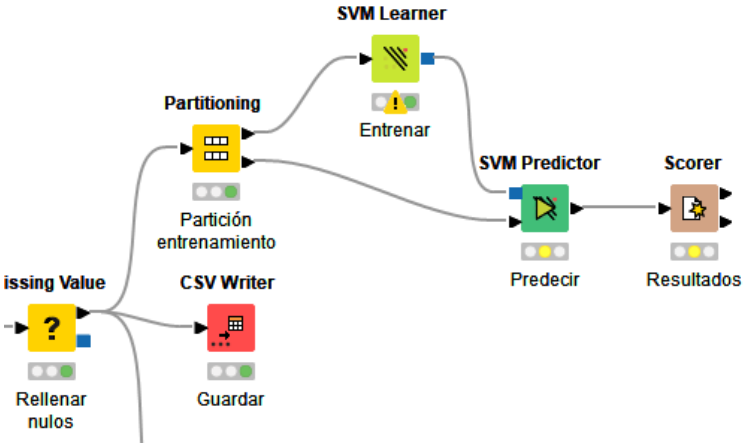
- ProcesarSalesRep:
    - Lee el fichero Sales Rep.xlsx.
    - Lee el fichero RegionDictionary.xlsx.
    - Usa Value Lookup y conecta las salidas de los 2 nodos anteriores a este. Configuración: Lookup column: (Sales Area) - Key column: (Misspelled) Lookup column output: (Replace) - Replacement column: (Regions)
    - Escribe el fichero excel resultante, incluyendo sus cabeceras y eligiendo sobrescribir si ya existe.

- CombinacionDatos:
    - Carga el CSV generado en el 1er workflow.

- Cambiar “Price Paid” a numérica, tendrás que reemplazar las comas decimales por puntos para poder hacer la conversión. Usa un String Replacer y luego String To Number.
- Lee el excel generado en el 2º workflow.
- Añade un “Joiner” con esta configuración:
  - Left Outer Join
  - Sales Area con Sales Area
  - Product Name con Product Name
  - Merge join columns
- Lee el fichero Product Info.xlsx.
- Reemplaza los valores vacíos de la columna Country con el valor de la fila anterior.
- Añade otro joiner para unificar la salida del 1er joiner con la salida del procesamiento de los productos, con esta configuración:
  - Left Outer Join
  - Country con Country
  - Product Name con Product Name
  - Merge join columns
- Limpieza de datos:
  - Eliminar filas duplicadas
  - Eliminar filas con “Quantity” 0 o negativo mediante Row Filter.
  - Eliminar filas con Product Name a #NV
  - Calcular beneficio por fila usando Math Formula. La operación es: precio pagado menos coste de producción unitario, todo ello multiplicado por la cantidad. Indicar que el cálculo se almacene en una columna llamada “Revenue”.
  - Guardar el resultado final en un fichero CSV.
  - En paralelo antes de escribir el CSV, obtén un gráfico circular de los beneficios acumulados por país.

Ejercicio 5 – Machine Learning

Partiendo del workflow del ejercicio 3, desde el nodo “Missing Value”, monta un flujo para entrenar y validar un modelo SVM cuyo target será “Survived”. Se adjunta el esquema para facilitar la tarea.



Cuando lo hayas conseguido, y partiendo del mismo dataset, resuelve la actividad, pero sustituyendo todo el flujo por un programa en un notebook de Jupyter ejecutado localmente. Recuerda que debes recrear todos los pasos desde el CSV Reader exactamente igual que están configurados en el flujo. Compara los resultados obtenidos y el tiempo de entrenamiento de cada caso.

Finalmente, realiza todo el proceso completo de preproceso de los datos del dataset del ejercicio, tal y como lo harías con lo aprendido en la unidad 3 (limpieza, sustitución, normalización, etc…). Cuando tengas los resultados compáralos con las otras 3 versiones. Reflexiona sobre esas diferencias y anótalo en un documento. Se pedirá en clase.

Ejercicio 6

Utiliza el dataset “cancer_mama.csv” que contiene información sobre la enfermedad para construir un Random Forest que prediga la columna objetivo “diagnosis” en función de las demás. Carga el CSV, elimina la primera columna (y la última si aparece vacía) y realiza el proceso necesario para conseguir el mejor modelo posible. Debes hacerlo íntegramente en Knime.

Ejercicio 7 – regresión lineal

Sigue estos pasos para crear un workflow para regresión lineal:

- Carga el archivo adult_joined.table ejecutando los nodos Table Reader y Missing Value.
- Divide los datos en un conjunto de entrenamiento (75 %) y un conjunto de prueba (25 %). Dividir aleatoriamente usando el nodo Partitioning.
- Entrena un modelo de regresión lineal en el conjunto de entrenamiento para predecir las horas de trabajo semanales. Utiliza todas las columnas excepto la columna "ID" para la predicción.
- Aplica el modelo al conjunto de prueba.
- Evalúa el rendimiento del modelo de regresión lineal con el nodo Numeric Scorer.

Ejercicio 8 – exploración de datos

Flujo A

1) Carga el fichero adult.csv

2) Inspecciona las propiedades de los datos con el nodo “Data Explorer”. Aviso: el nodo “Data Explorer” está en la extensión “KNIME JavaScript Views (Labs)”

- ¿Cuántas nacionalidades diferentes están representadas en los datos?

3) En la vista interactiva, excluye las columnas nominales que contienen valores faltantes.

- ¿Cuáles de las columnas nominales contienen valores faltantes? ¿Cuántos valores faltantes hay en cada una?

Flujo B

1) Carga el archivo adult_w_commute.table
2) Asigna colores a las filas según el estado civil.
3) Dibuja un diagrama de dispersión de horas semanales vs. tiempo de desplazamiento.
- ¿Observas alguna relación particular entre estas dos columnas?
- ¿Qué puedes decir sobre el estado civil de las personas que trabajan menos de 25 horas semanales y que tardan más tiempo en los desplazamientos para ir al trabajo?
4) Crea una tabla interactiva que muestre los datos. Usa el nodo “Table View”.
5) Encapsula los nodos de diagrama de dispersión y el “Table View” en un componente. Abre la vista interactiva del componente.
6) Cambia al modo "Seleccionar" del ratón en la vista del diagrama de dispersión y selecciona los puntos de datos con menos de 25 horas de trabajo y más de 120 minutos de desplazamiento.
7) Muestra solo las filas seleccionadas en la tabla interactiva.
- ¿Cuál es el país de origen de los puntos de datos seleccionados?

Ejercicio 9 – decisión tree

1) Carga el fichero adult_joined.table.
2) Separa los datos en “entrenamiento” y “test” en un 75-25, mediante muestreo estratificado a la columna income.
3) Entrena un modelo “Decision Tree” para predecir si una persona gana más de 50k anuales.
4) Valida en el conjunto de test.
5) Evalúa la precision del modelo mediante las métricas oportunas.
- ¿Cuál es la precisión general del modelo??
6) Abre la configuración del nodo Scorer (JavaScript) y excluye de la tabla de estadísticas de predicción de clase las estadísticas que también estén presentes en la matriz de confusión. Muestra el número de filas en la matriz de confusión.
- ¿Cuál es el número de filas en el conjunto de datos de prueba?
7) Evalúa el rendimiento del modelo con una curva ROC.
- ¿Cuál es el área bajo la curva para el modelo de árbol de decisión?
8) OPTIONAL: Try out other parameter settings to reach a better performance. For example, change the quality measure, pruning method, or minimum number of records.