



Práctica 01

EJERCICIOS PROPUESTOS.

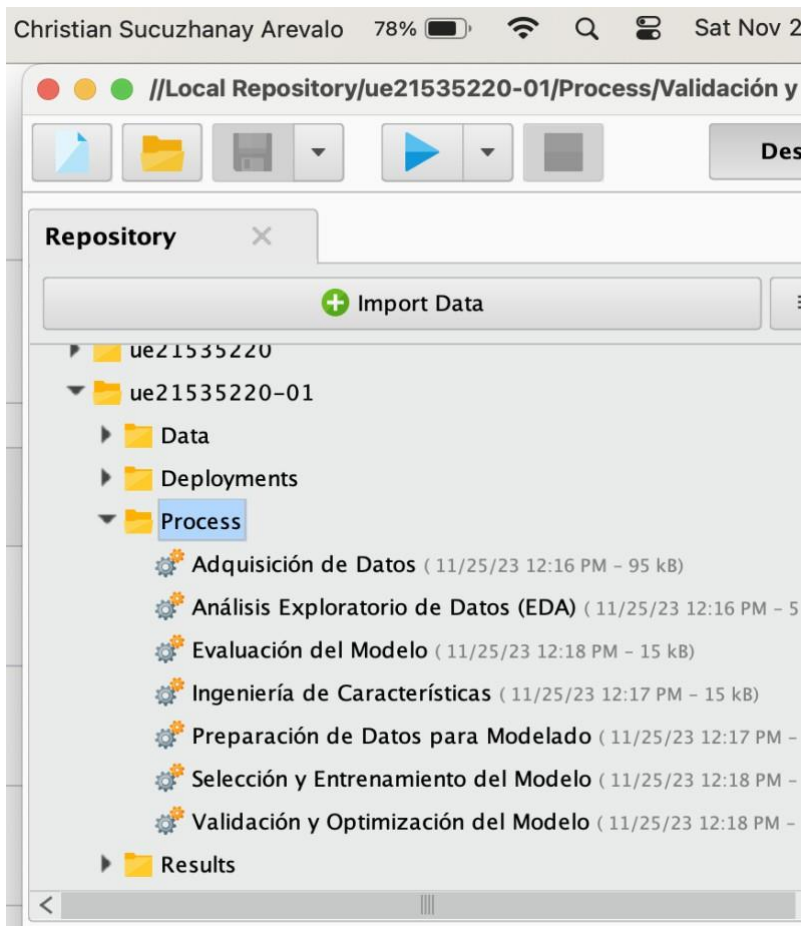
<u>LEER ANTES DE HACER NADA</u>	<u>3</u>
<u>EJERCICIOS PROPUESTOS</u>	<u>4</u>
<u>EJERCICIO 01</u>	<u>4</u>
<u>PREDICCIÓN DEL FRAUDE EN TARJETAS DE CRÉDITO</u>	<u>4</u>
INTRODUCCIÓN	4
OBJETIVO	4
PASOS PARA RESOLVER EL EJERCICIO	4
FRAUDE EN TARJETAS DE CRÉDITO	4
A. ADQUISICIÓN DE DATOS	4
B. ANÁLISIS EXPLORATORIO DE DATOS (EDA).....	6
C. INGENIERÍA DE CARACTERÍSTICAS.....	6
D. PREPARACIÓN DE DATOS PARA MODELADO.....	6
E. SELECCIÓN Y ENTRENAMIENTO DEL MODELO.....	6
F. EVALUACIÓN DEL MODELO	6
G. VALIDACIÓN Y OPTIMIZACIÓN DEL MODELO	7
DEMO FUNCIONAL	7
CONCLUSIONES	7
<u>EJERCICIO 02</u>	<u>8</u>
<u>CLASIFICACIÓN DE DOCUMENTOS</u>	<u>8</u>
INTRODUCCIÓN	8
ENTENDIENDO EL NEGOCIO	9
DESCRIPCIÓN DE LOS DATOS Y SU FUENTE	10
OBJETIVOS Y CRITERIOS DE ÉXITO	10
MODELADO	11
MODELO DE TEMAS	12





<u>A.</u>	<u>OBJETIVOS DE LA PRÁCTICA</u>	<u>13</u>
<u>A)</u>	<u>QUÉ DEBERÁ ENTREGAR / SUBIR AL CANVAS ¿?</u>	<u>13</u>
<u>B)</u>	<u>NOMBRADO DE ARCHIVOS E INDICACIONES DE CÓMO SUBIR Y FORMATO</u>	<u>13</u>

Leer antes de hacer nada



En esta práctica, debéis entregar resueltos los [ejercicios 01](#) y el [ejercicio 02](#), el 01 corresponde a la **predicción de tarjetas de crédito** y el segundo a **clasificación de texto**. En cada ejercicio se indica claramente donde está el dataset.

Para que la resolución sea exitosa y no se os **cuelgue** vuestros equipos, al resolver los ejercicios, debéis seguir los [“Pasos para RESOLVER el ejercicio”](#), estos pasos son el estándar en la resolución de problemas de ML, además define el esquema que debéis seguir para la resolución y entrega de la practica, es decir, estos pasos deben estar plasmados también en los

procesos de RapidMiner de vuestra entrega, como podéis observar en la captura adjunta.

[ejercicios 01](#)

EJERCICIOS PROPUESTOS.

Ejercicio 01

Predicción del Fraude en Tarjetas de Crédito

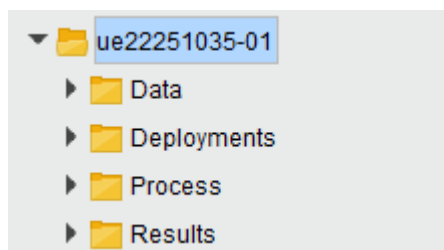
Introducción

La detección de fraude en transacciones con tarjetas de crédito es un desafío constante para las instituciones financieras. El uso de técnicas de machine learning puede ser fundamental para identificar patrones y anomalías en los datos que podrían indicar actividades fraudulentas.

Objetivo

Este informe describe los pasos esenciales, que realizaremos sobre el dataset (ver punto A) para predecir el fraude en transacciones con tarjetas de crédito utilizando técnicas de ML con diferentes herramientas, las mismas que se justifican por la premura en la entrega de la presente PRACTICA, entre otras :

1. RapidMiner



Pasos para RESOLVER el ejercicio:

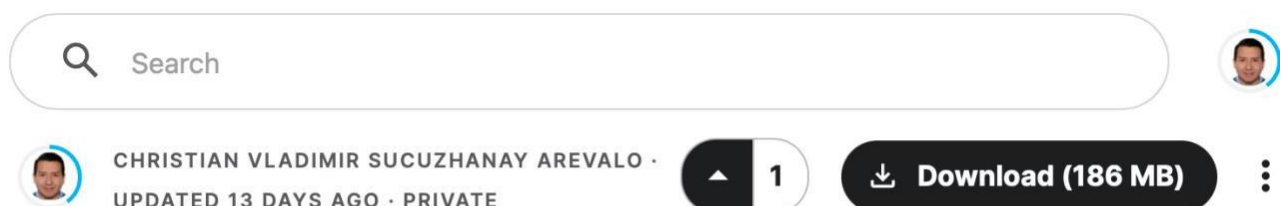
Fraude en Tarjetas de Crédito

A. Adquisición de Datos

Obtención de datos:

Os entrego un dataset con transacciones de tarjetas de crédito, no hay fechas en las transacciones, incluyen características como cantidad, tipo de transacción, entre otros.

Lo tenéis en mi perfil Kaggle:



Credit card fraud

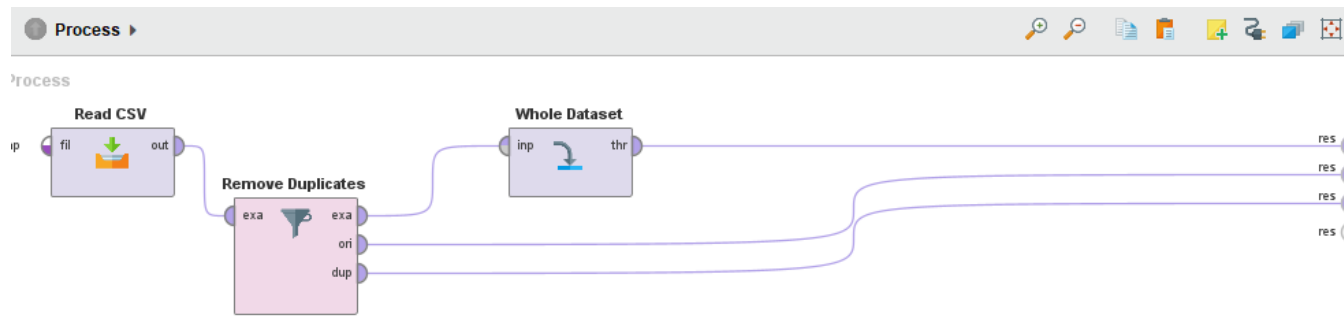
Descripción del dataset

- a) Nombre: full_dataset.csv
- b) Tipo: CSV
- c) Observaciones : 6.362.620 observaciones
- d) Atributos: 11 atributos.

Name	Type	Missing
✓ amount	Real	0
✓ oldbalanceOrg	Real	0
✓ newbalanceOrig	Real	0
✓ oldbalanceDest	Real	0
✓ newbalanceDest	Real	0
✓ step	Integer	0
✓ isFlaggedFraud	Integer	0
Label ✓ isFraud	Nominal	0
✓ type	Nominal	0
✓ nameOrig	Nominal	0
✓ nameDest	Nominal	0

Limpieza y preprocesamiento

- En primer lugar, llevamos a cabo la lectura de datos y procedemos a la eliminación de los registros no deseados. Los datos extraídos se almacenan en un archivo CSV llamado 'whole', que utilizaremos para realizar un análisis exploratorio de datos en etapas posteriores.



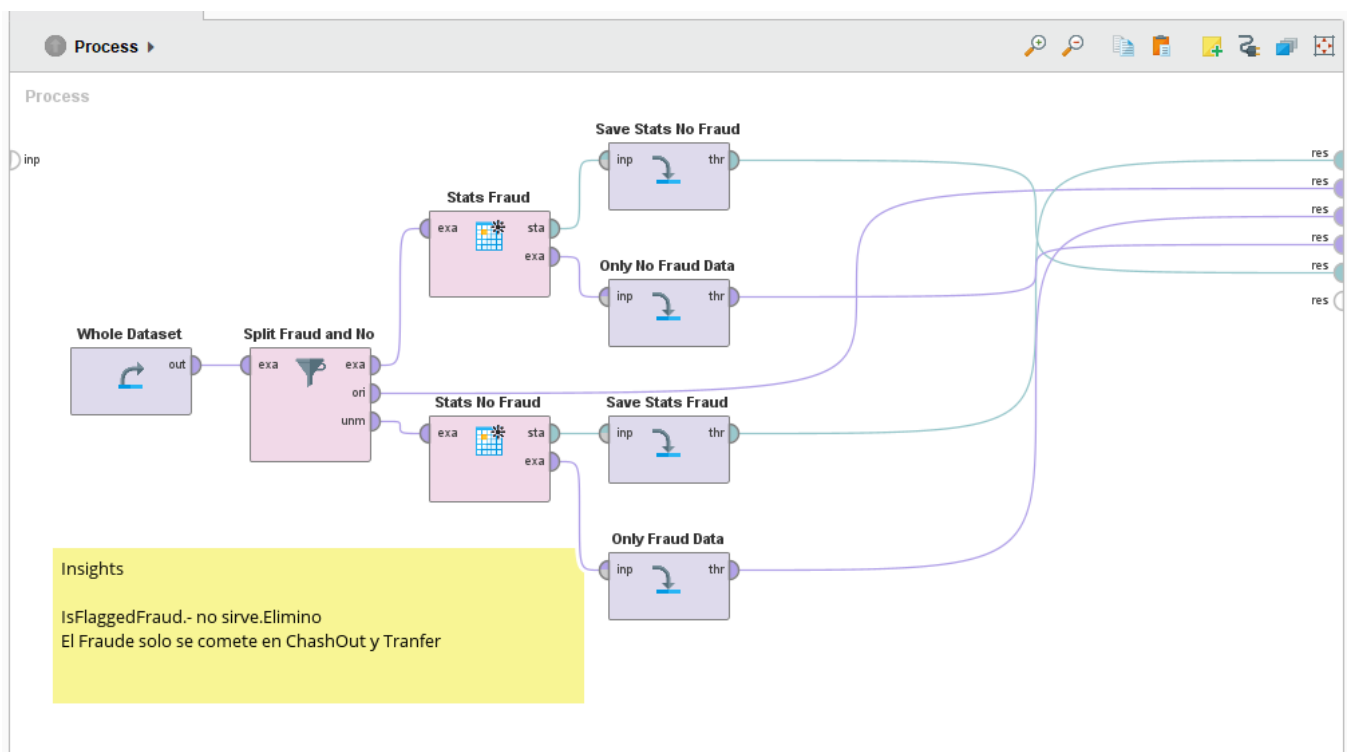
B. Análisis Exploratorio de Datos (EDA)

Durante la exploración de características, hemos determinado que la variable 'isFraud' es de particular interés, ya que indica si una transacción es fraudulenta o no. Además, en esta etapa llevamos a cabo las siguientes acciones:

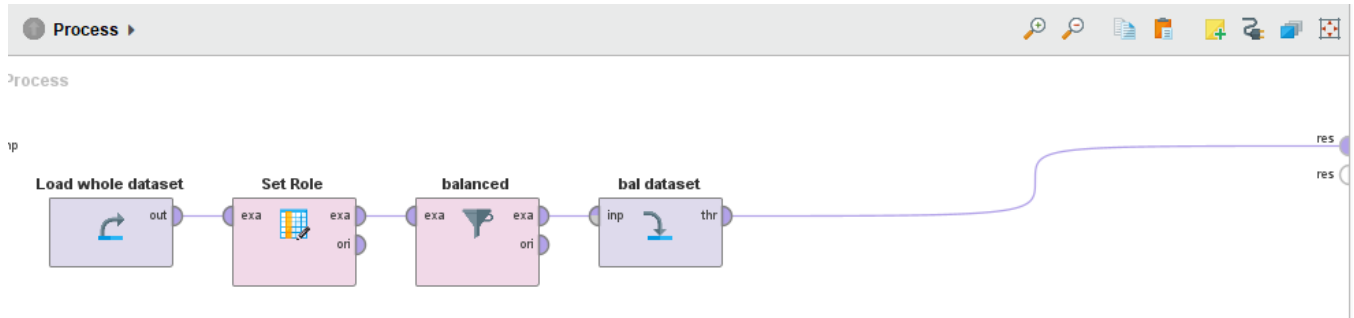
- Analizar la distribución de variables: Estudiamos cómo se distribuyen las diferentes variables en nuestro conjunto de datos y evaluamos su relevancia para la detección de fraudes.
- Identificar correlaciones: Buscamos relaciones o conexiones entre las variables, incluida 'isFraud', para comprender mejor cómo se relacionan entre sí y si pueden servir como indicadores de actividades fraudulentas.
- Búsqueda de patrones y anomalías: Exploramos el conjunto de datos en busca de posibles patrones que puedan indicar transacciones fraudulentas, así como anomalías que requieran una atención especial.

Además, utilizamos visualizaciones y gráficos para representar de manera efectiva la distribución de transacciones normales y fraudulentas, lo que nos ayuda a comprender mejor la naturaleza de los datos y a identificar posibles áreas de interés en nuestra detección de fraudes.

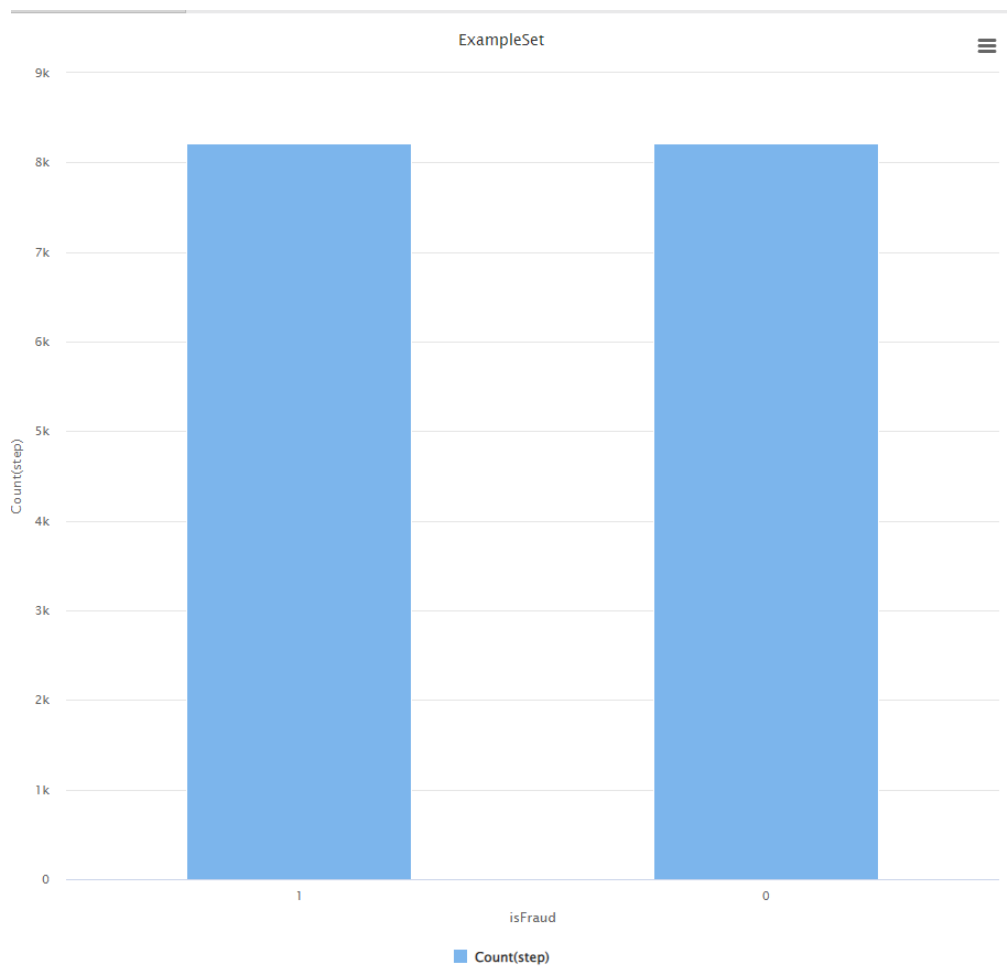
- **Análisis exploratorio de datos (EDA):**



- Balanceo de carga:

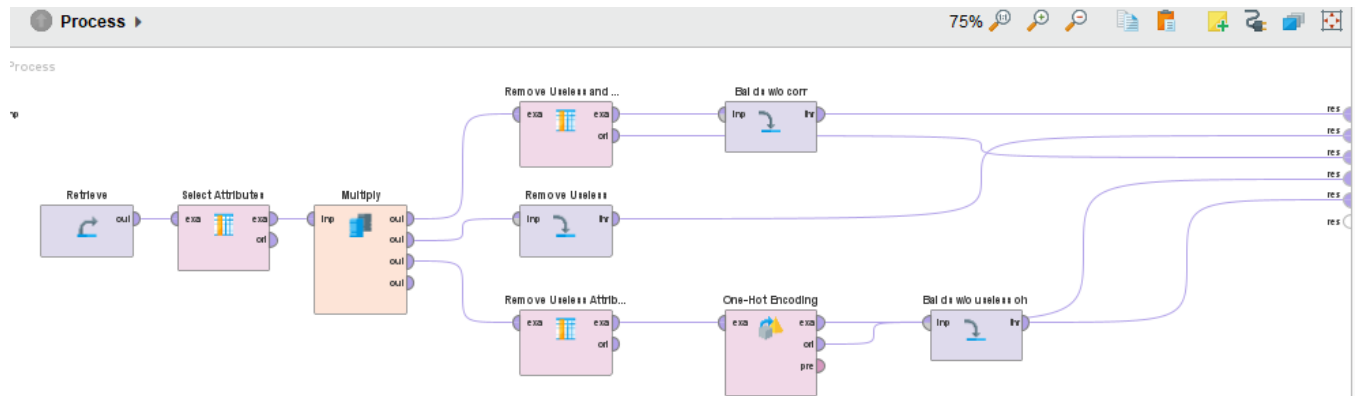


- Variable balanceada:



C. Ingeniería de Características

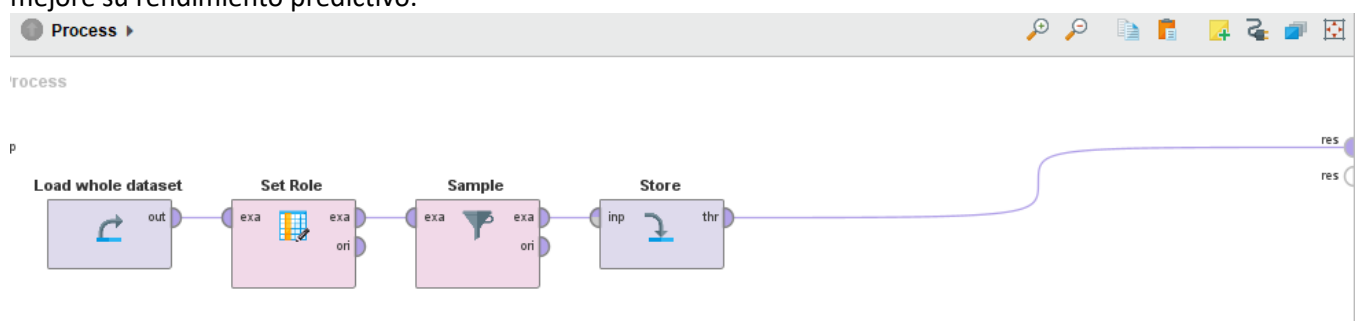
En la fase de selección de características, nuestro objetivo es identificar las variables más relevantes que contribuirán significativamente a nuestro modelo predictivo. Utilizamos técnicas de evaluación de características para determinar cuáles tienen un impacto significativo en la predicción de resultados.



Además, consideramos la creación de nuevas características mediante la derivación de variables adicionales a partir de las existentes. Estas nuevas características están diseñadas para capturar información importante que podría no estar representada de manera adecuada en las variables originales. La creación de nuevas características puede mejorar la capacidad predictiva del modelo al proporcionarle más información relevante para tomar decisiones precisas.

D. Preparación de Datos para Modelado

Preparamos y ajustamos el conjunto de datos, donde los datos se dividen en conjuntos para entrenamiento, validación y prueba. Además, se aborda el balanceo de clases mediante técnicas como el undersampling, que se ha aplicado previamente, y el oversampling, que se planea implementar más adelante con recursos adicionales en la nube, para asegurar que el modelo de aprendizaje automático maneje las clases de manera equitativa y mejore su rendimiento predictivo.

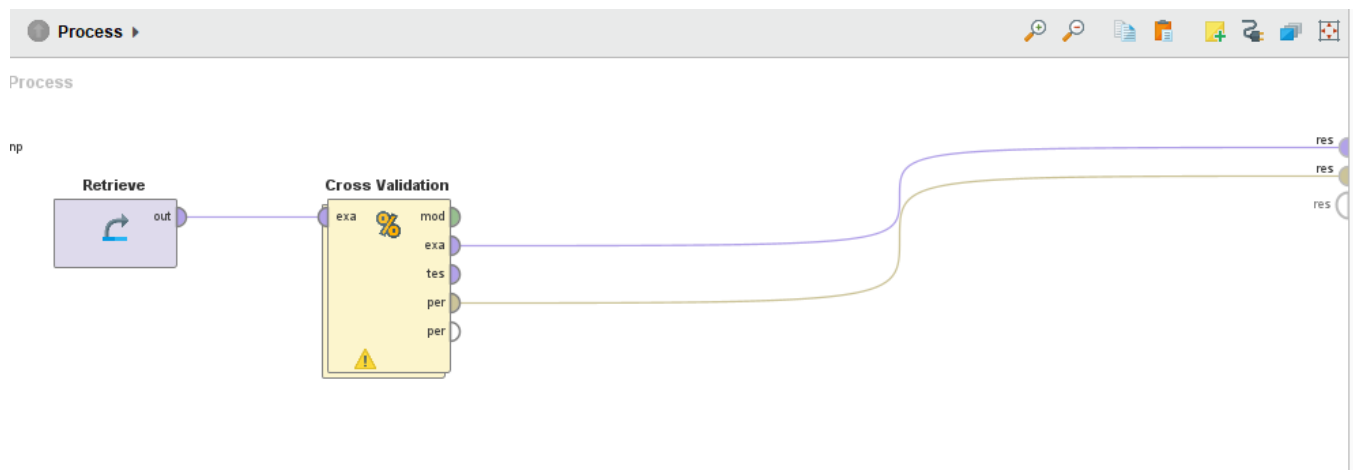


E. Selección y Entrenamiento del Modelo

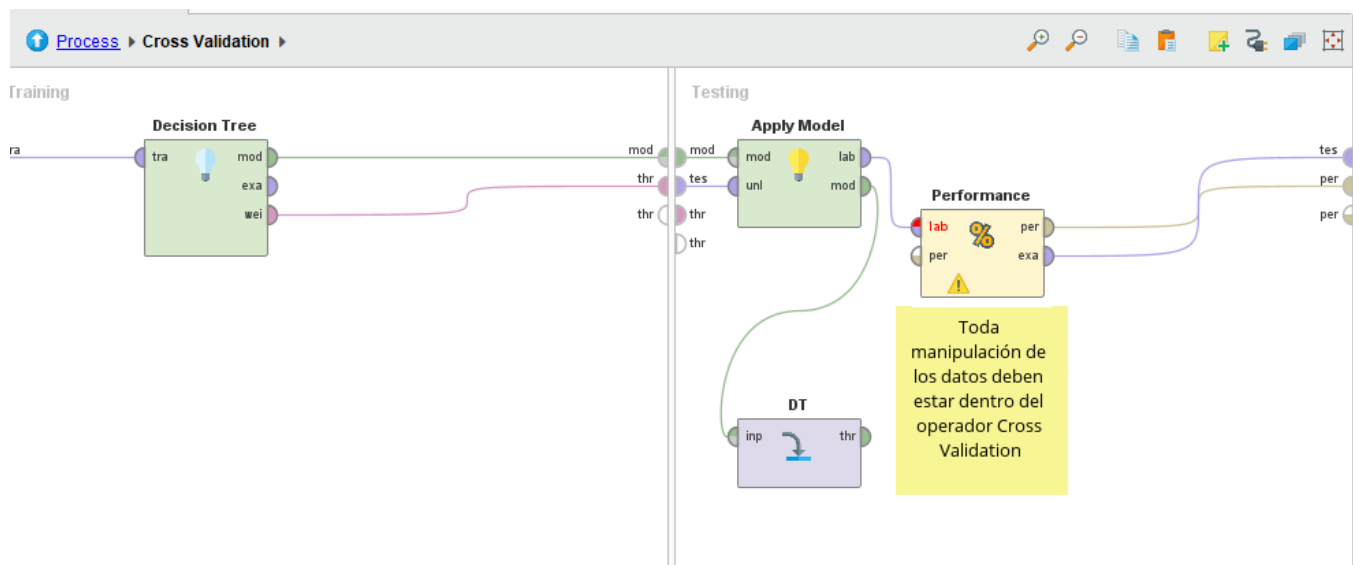
Elección del algoritmo: Seleccionar modelos de machine learning adecuados para la predicción de fraude (Random Forest, Support Vector Machines, Redes Neuronales, etc.).

Entrenamiento del modelo: Utilizar el conjunto de entrenamiento para entrenar el modelo seleccionado.

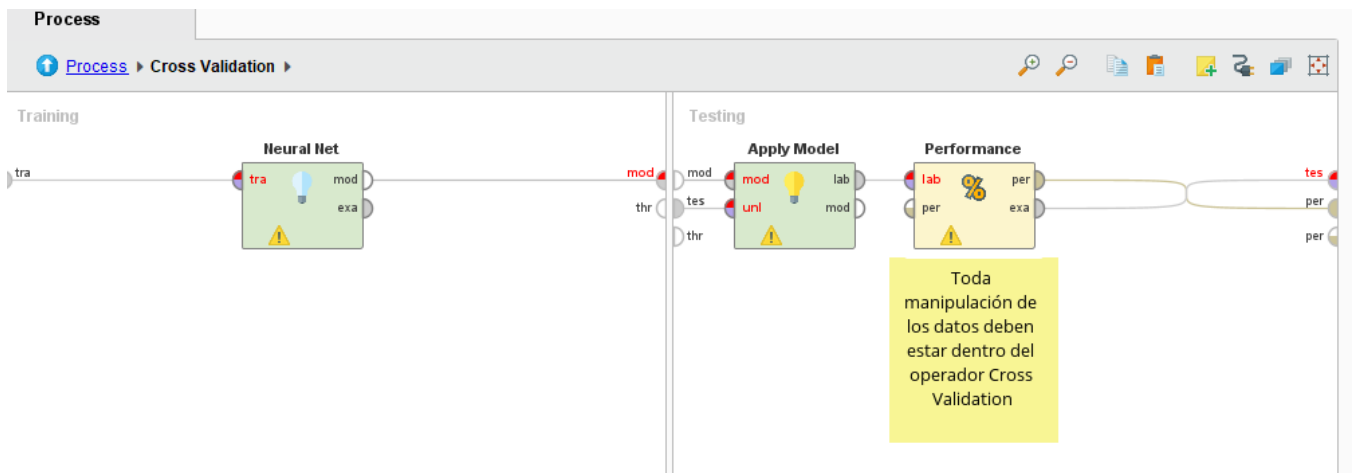
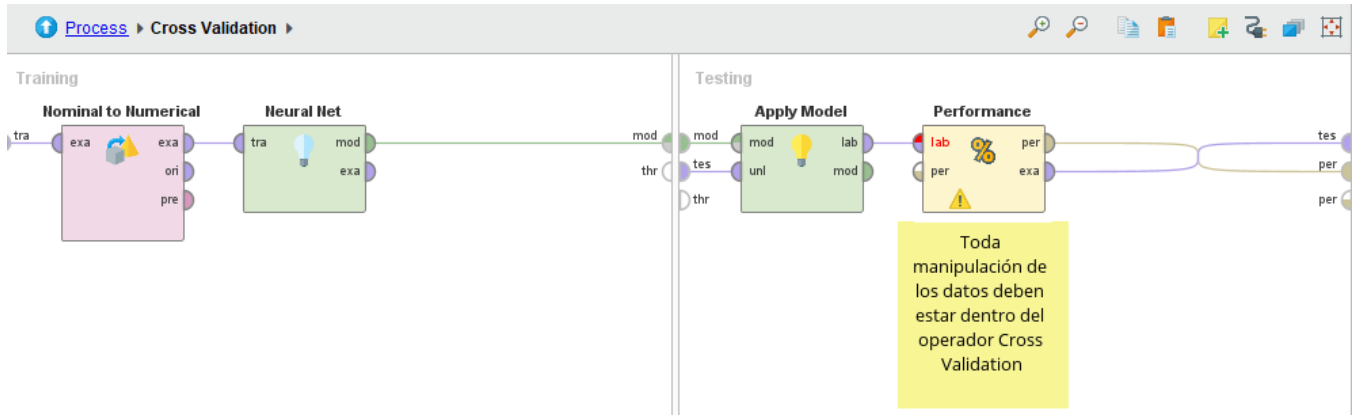
- Cross Validation



- Decisión Tree



- Neural Net



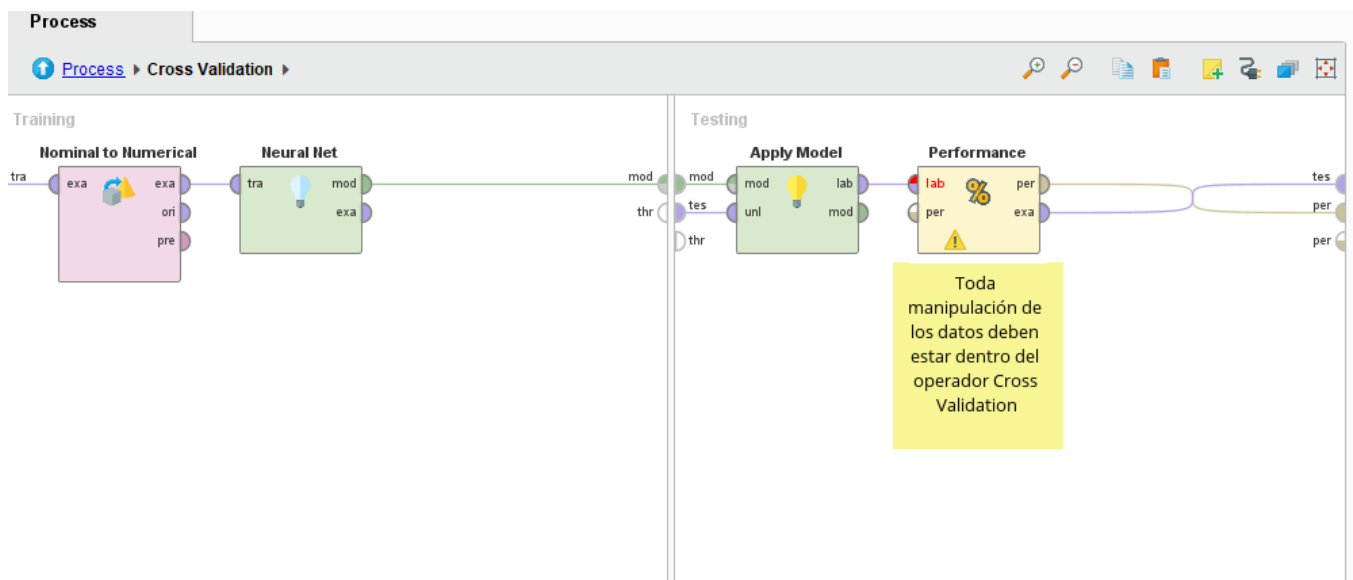
F. Evaluación del Modelo

En el contexto de la detección de fraude para nuestra variable objetivo isfraud, la evaluación realizada en RapidMiner ha revelado que una red neuronal es el modelo más adecuado. Este resultado se basa en una serie de pruebas exhaustivas, donde la red neuronal ha superado consistentemente a otros modelos en términos de precisión.

La métrica de precisión, que mide la proporción de predicciones correctas de fraude entre todas las clasificaciones de fraude realizadas por el modelo, es de vital importancia en el ámbito financiero. Un alto porcentaje de precisión implica que la red neuronal ha logrado distinguir con éxito las transacciones legítimas de las fraudulentas, minimizando los costosos errores de clasificación.

La red neuronal implementada fue cuidadosamente ajustada a través de un meticuloso proceso de selección de hiperparámetros y características, guiado por las capacidades analíticas de RapidMiner. Este enfoque iterativo ha permitido optimizar el modelo para capturar las complejidades y patrones no lineales que caracterizan el comportamiento fraudulento.

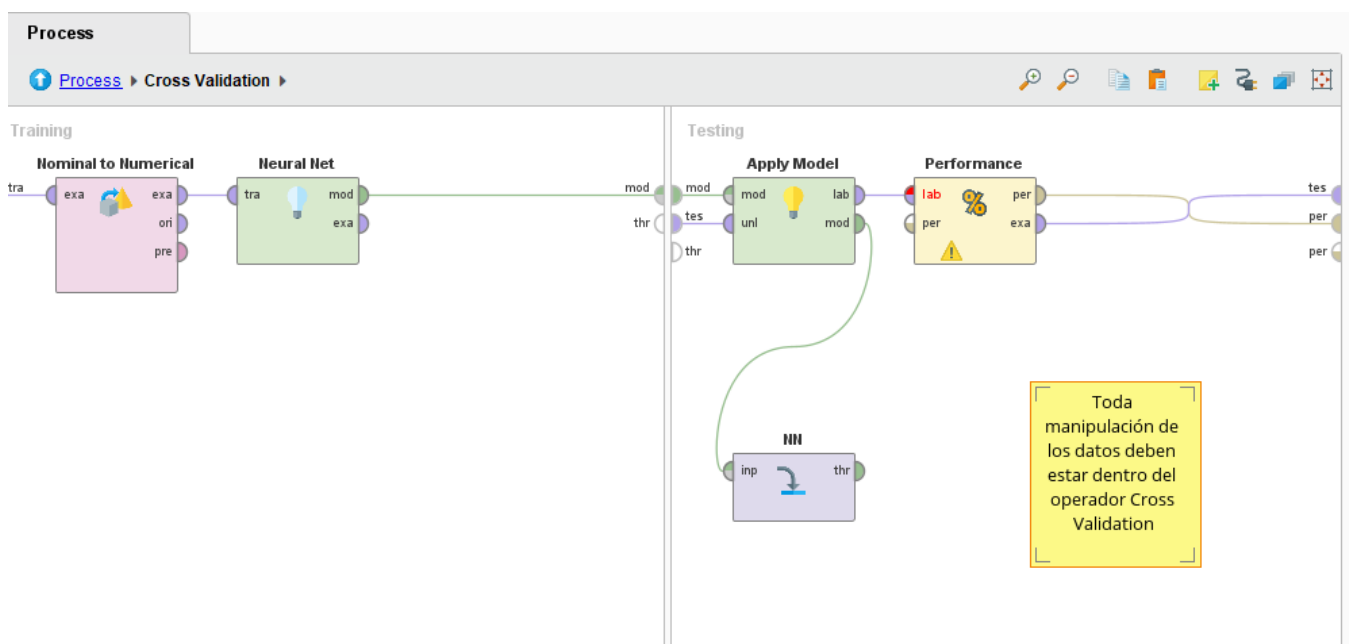
Los resultados obtenidos de la validación cruzada dentro de RapidMiner han demostrado de manera consistente que la red neuronal proporciona el mejor equilibrio entre sensibilidad y especificidad, ofreciendo un rendimiento superior en la clasificación precisa de las transacciones como fraudulentas o no fraudulentas. Estas evidencias estadísticas refuerzan nuestra decisión de adoptar la red neuronal como la solución principal para la prevención de fraude en nuestro sistema.



G. Validación y Optimización del Modelo

La validación cruzada es una técnica fundamental en el desarrollo de modelos de Machine Learning, ya que permite evaluar la capacidad de generalización de un modelo más allá del conjunto de datos de entrenamiento. Al dividir el conjunto de datos en varias particiones y utilizar sucesivamente una de ellas como conjunto de prueba y el resto como entrenamiento, podemos obtener una estimación más fiable de cómo el modelo se comportará ante datos nuevos y desconocidos. Tras esta etapa, es común realizar una optimización adicional del modelo, ajustando hiperparámetros y refinando características para mejorar aún más su precisión y capacidad predictiva. Este proceso iterativo de validación y optimización es crucial para desarrollar un sistema de detección de fraude robusto y confiable.

Repositorio del código



<https://github.com/jafethsuarez/sistemasInteligentes.git>

Conclusiones

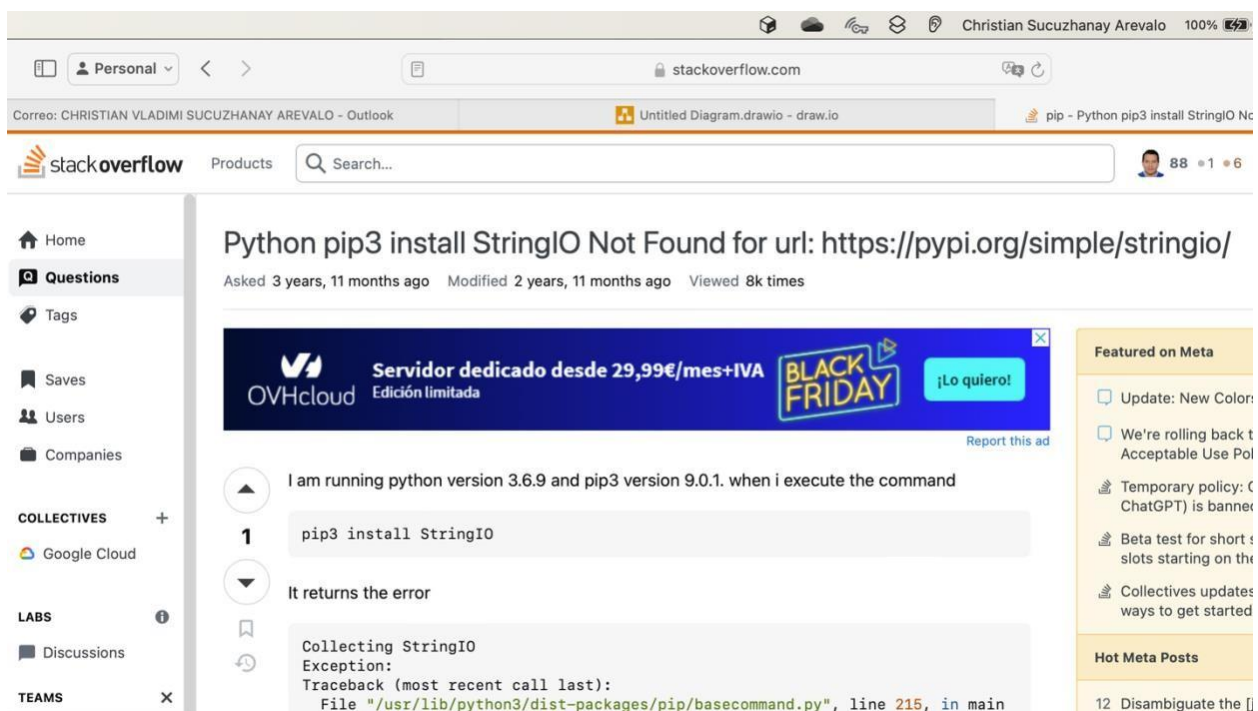
La implementación de modelos de Machine Learning para la detección de fraude en tarjetas de crédito representa un pilar clave en la mitigación de riesgos financieros. A pesar de que el sistema descrito se basa en una arquitectura sólida para el análisis de datos estáticos, para operar efectivamente en el ámbito de transacciones en tiempo real, se requiere la incorporación de componentes adicionales que permitan el procesamiento instantáneo de datos y la adaptación continua del modelo. Esta evolución arquitectónica es esencial para mantener la relevancia y efectividad del sistema frente a las dinámicas cambiantes del fraude financiero.

Ejercicio 02

Clasificación de documentos






Introducción

Usaremos un dataset de Stackoverflow, donde hay diferentes categorías, las mismas albergan preguntas de usuarios sobre diferentes temas.



The screenshot shows a web browser window with the URL `stackoverflow.com`. The page displays a question titled "Python pip3 install StringIO Not Found for url: https://pypi.org/simple/stringio/". The question was asked 3 years, 11 months ago, modified 2 years, 11 months ago, and viewed 8k times. The question text is: "I am running python version 3.6.9 and pip3 version 9.0.1. when i execute the command `pip3 install StringIO` It returns the error". The error message is: "Collecting StringIO\nException:\nTraceback (most recent call last):\nFile "/usr/lib/python3/dist-packages/pip/basecommand.py", line 215, in main". The page also features an advertisement for OVHcloud and a sidebar with navigation links like Home, Questions, Tags, Saves, Users, Companies, COLLECTIVES, LABS, Discussions, and TEAMS.

Entendiendo el negocio

88● 1 ● 6

Unix

Me da error al instalar usando Pip y me aparece los siguiente

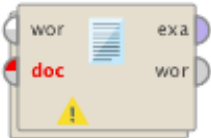
Linux


Me da error al instalar usando Pip y me aparece los siguiente


Android

Me da error al instalar usando Pip y me aparece los siguiente

Process Documents








Descripción de los datos y su fuente.

Los datos se obtendrán de Kaggle, y corresponden a preguntas de usuarios, de Stack Overflow para temas determinados. Lo tenéis en mi perfil de Kaggle



Stackoverflow Questions

▲ 0




Download (2 MB)

Data Card

Code (0)

Discussion (0)

Settings

full-dataset.csv (6.61 M...   

Detail


Compact

Column

3 of 3 columns ▼

▲ FullText	▲ TicketType	▲ Po
6599 unique values	Unix Apple Other (3119)	30% 23% 47%

Version 1 (6.61 MB)

 full-dataset.csv

Summary

- 1 file
- 3 columns

Objetivos y criterios de éxito

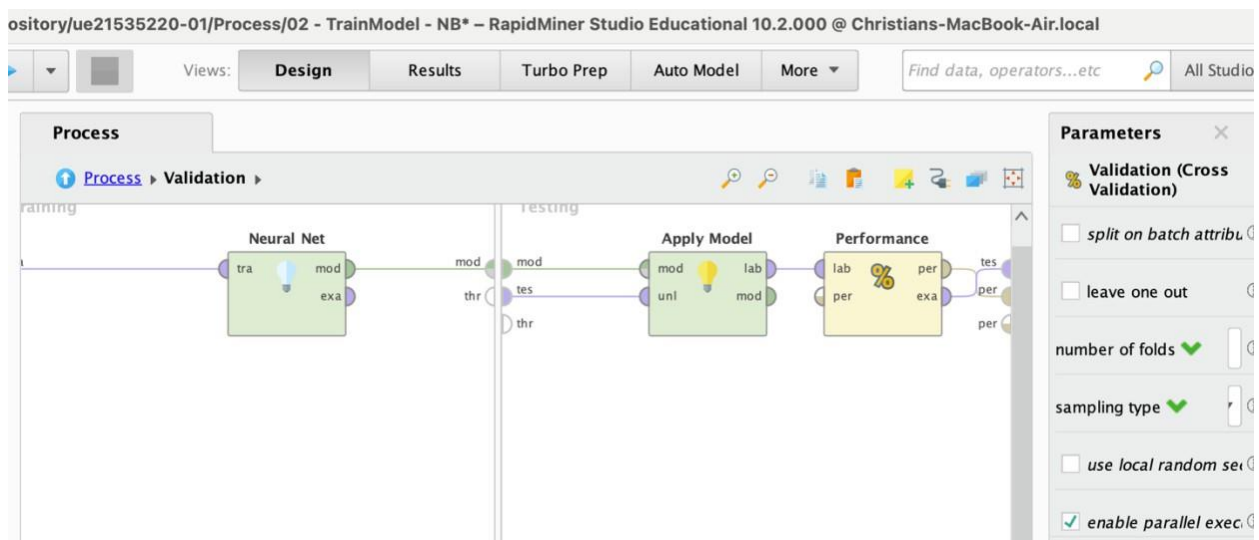
El objetivo principal es proporcionar una herramienta que automatice la clasificación de las preguntas de los usuarios de StackOverflow.

Debéis alcanzar mínimo **93%**.

El que más precisión alcance tendrá diez y a partir de ahí irá bajando la nota

Modelado

Necesitamos construir y validar una clasificación multinivel para n-clases. La idea básica aquí es que cada clase debe tener un conjunto único de palabras clave asociadas con ese tipo de clase. Por ej. Los términos utilizados para los dispositivos Apple deberían ser diferentes para los dispositivos Android y los algoritmos de aprendizaje automático deben poder diferenciarlos, quieres paso ejemplo del porcentaje de predicción que yo he obtenido sin realizar ningún proceso adicional.



accuracy: 86.71% +/- 0.87% (micro average: 86.71%)

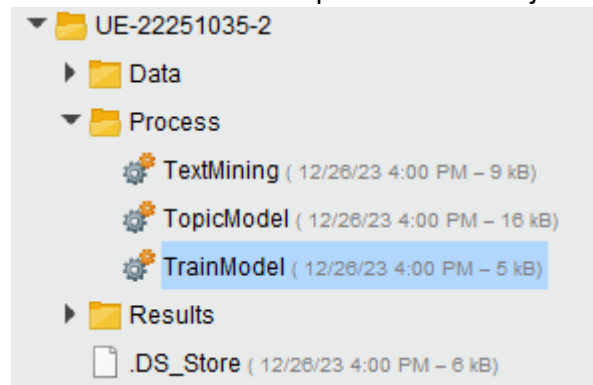
	true 3DPrinting	true Android	true Apple	true Dba	true Unix	class precision
pred. 3DPrinting	1184	7	12	2	26	96.18%
pred. Android	12	1578	196	1	147	81.59%
pred. Apple	19	147	2076	13	242	83.14%
pred. Dba	9	6	29	2017	202	89.13%
pred. Unix	26	62	187	117	2683	87.25%
class recall	94.72%	87.67%	83.04%	93.81%	81.30%	

Modelo de temas

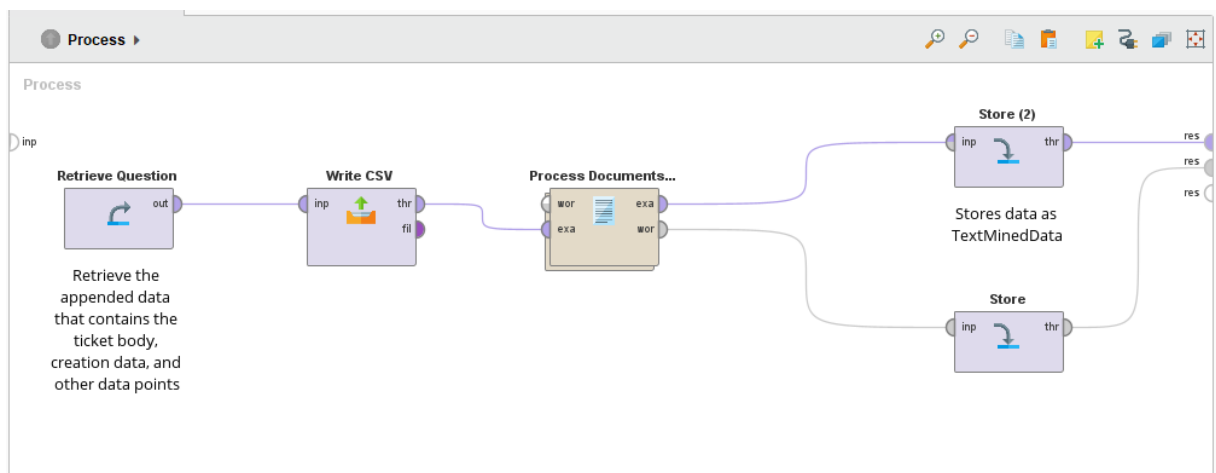
Conseguido lo anterior, debemos discernir cuál es el problema en la pregunta para poder darle una solución. Actualmente, solo tenemos las publicaciones en sí y no tenemos categorías para estas publicaciones. Necesitamos crear una lista de temas para cada pregunta, para que los equipos puedan asignar recursos, personal, técnicos de manera efectiva para resolverlas

Este ejercicio lo debéis realizar exactamente igual, siguiendo los pasos del ejercicio 01, realizando la misma estructura de folders en Rapidminer

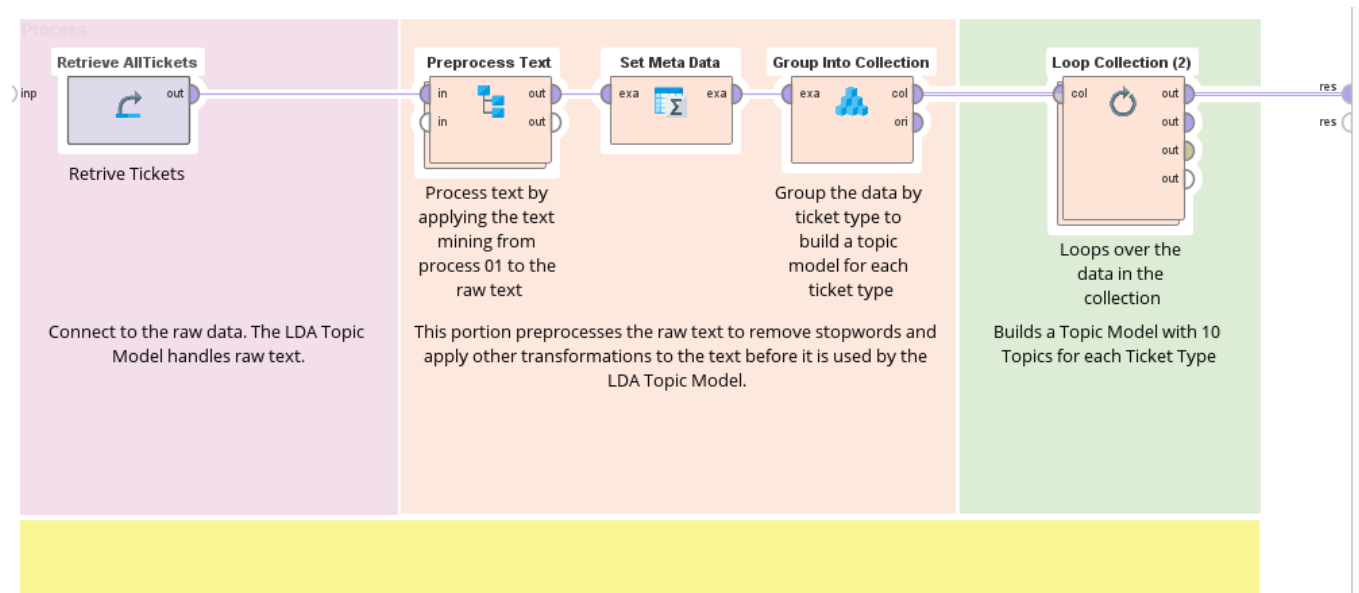
- Definimos nuestra estructura con los procesos a trabajar:



- **Proceso TextMining:** Presentamos un sistema de procesamiento de datos que inicia con la extracción de información a través del componente "Retrieve Question". Este paso implica la recuperación de datos estructurados, incluyendo detalles críticos como el cuerpo del ticket y metadata asociada. Posteriormente, estos datos se transcriben en un formato CSV por medio del componente "Write CSV", que facilita la manipulación y el análisis posterior. A continuación, se lleva a cabo un procesamiento de texto más profundo, indicado por el componente "Process Documents", cuyo propósito es refinar los datos para la extracción de características relevantes. El proceso concluye con el almacenamiento de los datos procesados, designados como "TextMinedData", junto con una copia general en el sistema de almacenamiento, indicado por el componente "Store". Este procedimiento asegura que los datos no solo se recopilan y estructuran adecuadamente, sino que también se preparan meticulosamente para análisis subsiguientes.

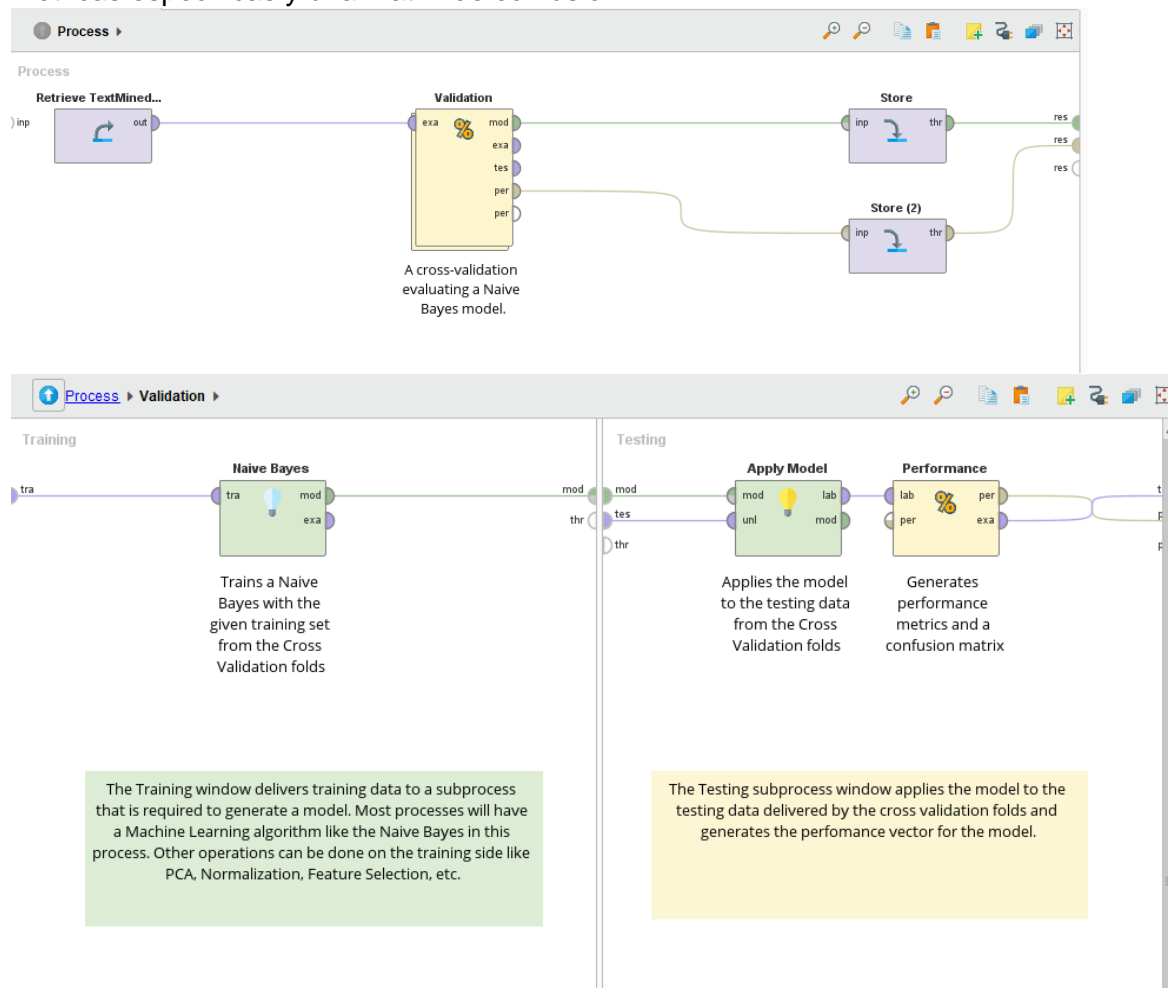


- Proceso TopicModel: Análisis de texto basado en un Modelo de Tópicos LDA.** El proceso inicia con la recolección de todos los tickets mediante "Retrieve AllTickets", lo cual indica una conexión directa con la base de datos de textos sin procesar. Seguidamente, se efectúa un preprocesamiento textual, removiendo palabras irrelevantes y realizando transformaciones necesarias para afinar los datos para el análisis de tópicos. Los datos preprocesados se clasifican por tipo de ticket en el paso "Set Meta Data" y posteriormente se agrupan por categorías en "Group Into Collection". Finalmente, la iteración sobre estas colecciones se realiza mediante "Loop Collection", donde se construye un modelo LDA para discernir 10 tópicos distintos dentro de cada tipo de ticket. Este método permite identificar patrones y temas subyacentes en los datos, lo cual es esencial para la comprensión detallada y la categorización del contenido de los tickets.



- Proceso TrainModel:

Validación y el entrenamiento de un modelo de Naive Bayes en un entorno de Machine Learning. Inicia con la recuperación de datos preprocesados para texto, presumiblemente listos para el análisis de Machine Learning, seguido por un paso de validación cruzada que evalúa la generalización y efectividad del modelo de Naive Bayes. Tras la validación, los resultados del modelo son almacenados, indicando posiblemente la retención de información sobre el rendimiento del modelo. El segundo diagrama detalla aún más el proceso, mostrando que el modelo de Naive Bayes es entrenado con datos de entrenamiento derivados de los pliegues de la validación cruzada, y luego se aplica a los datos de prueba para evaluar su rendimiento mediante métricas específicas y una matriz de confusión.



- Matriz de confusión

El análisis de la matriz de confusión para nuestro modelo de clasificación muestra una precisión general notable de 86.64% con un margen de error del $\pm 1.28\%$, lo que indica una robustez considerable en la clasificación de las categorías establecidas: 3DPrinting, Android, Apple, DbA y Unix. La precisión por clase, que refleja la proporción de predicciones correctas para cada categoría, destaca particularmente en la clase 3DPrinting con un 96.05%. Por otro lado, el recall por clase, que mide la capacidad del modelo para detectar todas las instancias relevantes de una categoría, sobresale también en 3DPrinting con un 95.20%. Estos resultados demuestran la eficacia del modelo en la clasificación precisa de las categorías, aunque también revelan ciertos desafíos, como se observa en las confusiones entre categorías, tales como los casos de Apple erróneamente clasificados como Android y viceversa. Este nivel de detalle es crucial para identificar específicamente dónde el modelo puede ser afinado para mejorar su capacidad de clasificación.

accuracy: 86.64% +/- 1.28% (micro average: 86.64%)

	true 3DPrinting	true Android	true Apple	true DbA	true Unix	class precision
pred. 3DPrinting	1190	9	12	4	24	96.05%
pred. Android	11	1576	192	1	144	81.91%
pred. Apple	17	140	2073	14	259	82.82%
pred. DbA	8	5	23	2007	190	89.88%
pred. Unix	24	69	200	124	2683	86.55%
class recall	95.20%	87.60%	82.92%	93.35%	81.30%	



A. Objetivos de la Práctica:

1. Conocer como se realiza un ETL + EDA + Modeling y demás pasos que se citan en el apartado: [Pasos para RESOLVER el ejercicio](#)
2. Aplicar la plataforma RapidMiner para resolver preguntas de negocio.
3. Entender cómo funcionan los modelos y elegir el de mejor rendimiento
4. Generar el documento que deberá ser subido al Canvas
5. Dejar la base sólida para pasar a los sistemas recomendación, crear construir la gente inteligente y general el producto final con el que culminamos la asignatura

a) Qué deberá entregar / subir al CANVAS ¿? :

- a. Memoria descriptiva PDF, con portada, índice de la actividad, donde figuren las respuestas a las preguntas del enunciado, así como todos los gráficos / mapas / informes generados, capturas. (debe constar el nombre del alumno, con su enlace al repositorio donde esta el código entregado, **IMPORTANTE** el enlace debe estar activado (hyperlink).
- b. Se deberá explicar los procesos y decisiones tomadas y el porque, caso contrario, la actividad se considera no entregada.
- c. Todo el repositorio deberá subirse al repositorio de GitHub de cada alumno y añadirme como colaborador (con el rol de propietario para poder evaluar) al repositorio de vuestro GitHub; mi username es : sukuzhanay@gmail.com (**si no me añadís, se considera no entregado**)
- d. La entrega es individual
- e. **IMPRESINDIBLE** : los archivos deberán ser subidos a vuestros repositorios .
- f. Fecha de entrega: según figure en el Canvas.

b) Nombrado de archivos e indicaciones de cómo subir y formato:

- a. Todos los archivos entregados deberán subirse de forma individual, NO comprimidos (zip, tar, etc)