GUÍA DE EVALUACIÓN UNIDAD IV

DATOS GENERALES DEL PROCESO DE EVALUACIÓN

Nombre(s) del participante expositor: ROMÁN OMAR FISCAL POLITO	Equipo: 2
Tema de Exposición: RapidMinner	Fecha: 17/NOV/2025
Asignatura: ECOSISTEMAS DE BIG DATA	Período Semestral:
Nombre del Docente: JUAN RAFAEL GONZÁLEZ CADENA	

INSTRUCCIONES

Revisar los documentos o actividades solicitadas y marque en los apartados “SI” cuando la evidencia a evaluar se cumple; caso contrario marque “NO”. En la siguiente columna “JUSTIFICACIÓN” ocúpela para explicar el porqué sí o no se cumple el reactivo.

	Características a cumplir (Reactivo)	Cumple SI	Cumple NO	OBSERVACIONES
Diapositivas y formato general				
1	Consideras que la presentación tiene una portada adecuada?	**X**		
1	Consideras que la presentación muestra letra apropiada	**X**		
Ortografía y redacción				
1	¿Consideras que el nivel de ortografía es aceptable?	**X**		
1	¿Consideras que la redacción es adecuada?	**X**		
Sobre la exposición y las formas				
1	¿Consideras que la presentación sigue un orden específico?	**X**		
1	¿La seguridad que demuestra es apropiado?, o recomendarías que trabaje en ello?	**X**		
Sobre el contenido				
4	¿La investigación desarrollada es seria y te proporciona seguridad para citarla o referenciarla?	**X**		
	CALIFICACIÓN	**10%**		

ING. INFO. 910-A

INSTITUTO TECNOLÓGICO SUPERIOR DE SAN ANDRÉS TUXTLA

Carrera: Ing. en Informática

Catedrático: M.T.I. Juan Rafael González Cadena

Semestre: 9no. Semestre

Grupo: 910-A

Alumno:
Román Omar Fiscal Pólito
Christian Manuel Millán Pólito

Periodo Escolar: Agosto-Diciembre 2025

San Andrés Tuxtla, Ver.

ING. INFO. 910-A

Plataforma líder de ciencia de datos y minería de información para análisis predictivo empresarial

Del Laboratorio Universitario al Líder Global

1 — 2001: Nacimiento

Creado como YALE en la Universidad Técnica de Dortmund, Alemania

2 — 2006–2007: Rebranding

Transición a RapidMiner bajo la empresa Rapid-I

3 — 2022: Nueva Era

Adquisición por Altair Engineering

4 — Hoy: Líder del Mercado

Plataforma consolidada en ciencia de datos empresarial

De proyecto académico a solución empresarial reconocida mundialmente, RapidMiner ha recorrido más de dos décadas de innovación continua en análisis de datos.

¿Qué es RapidMiner?

RapidMiner es una plataforma integral de ciencia de datos que democratiza el análisis predictivo mediante un entorno visual intuitivo.

Interfaz Visual

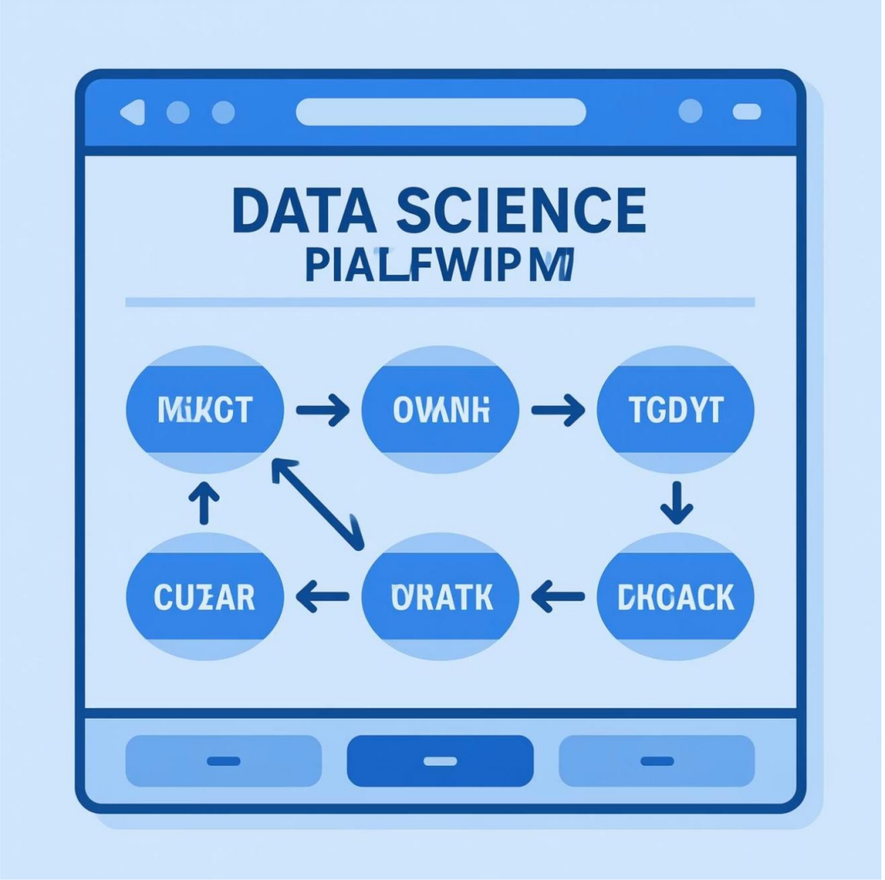
Diseño tipo arrastrar y soltar sin necesidad de programación extensiva

Flujo Completo

Desde preparación de datos hasta despliegue de modelos

Multiplataforma

Desarrollado en Java con arquitectura cliente-servidor



Arquitectura y Características Principales

1

Procesos Visuales

Workflows basados en operadores que ejecutan tareas de forma secuencial y conectada

2

Librería Extensa

Cientos de operadores predefinidos para limpieza, transformación, modelado y visualización de datos

3

Integración Múltiple

Conexión con Excel, bases SQL, archivos planos y extensiones R/Python para funcionalidades avanzadas

4

Capacidades Modernas

Analítica escalable, integración con IA y procesamiento de datos oscuros no estructurados

Casos de Uso en la Industria

Marketing Inteligente

Segmentación de clientes y análisis de comportamiento para campañas personalizadas

Detección de Fraudes

Identificación de transacciones sospechosas en finanzas y seguros con precisión

Mantenimiento Predictivo

Anticipación de fallos en equipos industriales mediante análisis de datos operativos

Educación e Investigación

Entorno ideal para prototipado rápido y enseñanza de ciencia de datos

Impacto Global en Números

400K+

Usuarios Activos

Profesionales y organizaciones en todo el mundo

20+

Años de Evolución

Desde proyecto académico hasta líder empresarial

Top 5

Ranking Industrial

Entre las herramientas más populares de minería de datos

Empresas Fortune 500 y organizaciones líderes confían en RapidMiner para sus iniciativas de análisis predictivo y transformación digital basada en datos.

Ventajas Competitivas

Democratización del Análisis

Usuarios sin perfil técnico pueden crear flujos complejos sin escribir código extensivo

Velocidad de Desarrollo

De datos crudos a insights accionables en tiempo récord mediante automatización inteligente

Ecosistema Robusto

Comunidad activa, plugins extensivos e integración flexible con R y Python

Escalabilidad Empresarial

Versión corporativa con soporte completo para despliegue en producción

Consideraciones y Limitaciones

Limitaciones de la Versión Gratuita

Restricciones en volumen de datos procesables y número de procesadores lógicos disponibles para ejecución

Requisitos de Infraestructura

Escenarios de Big Data o producción de alta demanda requieren hardware potente y configuración optimizada

Dependencia de Calidad de Datos

La confiabilidad de resultados depende críticamente de la calidad del input y del diseño del proceso analítico

Desafíos de Equidad Algorítmica

Plataformas AutoML pueden heredar sesgos; requiere validación rigurosa y prácticas éticas conscientes

Como toda herramienta poderosa, RapidMiner requiere uso responsable y comprensión de sus limitaciones técnicas y éticas.

Confiabilidad y Buenas Prácticas

Fortalezas de Confiabilidad

- Transparencia en modelos con capacidades de explicabilidad integradas
- Adopción por empresas Fortune 500 para decisiones críticas de negocio
- Herramientas de validación y evaluación de modelos incorporadas

Recomendaciones Esenciales

- Implementar validación cruzada rigurosa de modelos
- Monitorear métricas de sesgo y equidad algorítmica
- Realizar pruebas exhaustivas antes de despliegue en producción
- No confiar únicamente en automatización sin revisión humana experta

Principio clave: La automatización acelera, pero la supervisión experta garantiza resultados confiables y éticos en ciencia de datos.

Próximos Pasos: Tu Viaje con RapidMiner

01

Explora la Plataforma

Descarga la versión gratuita y familiarízate con la interfaz visual

02

Comienza con un Proyecto Pequeño

Importa un dataset sencillo y experimenta con flujos básicos de limpieza y modelado

03

Aprende de la Comunidad

Accede a tutoriales, documentación oficial y casos de uso compartidos

04

Evalúa Aplicaciones en tu Organización

Identifica casos de uso específicos donde RapidMiner pueda aportar valor analítico

Recurso recomendado: Tutorial completo para principiantes disponible en YouTube que cubre desde importación de datos hasta visualización de resultados.

¿Preguntas? Estamos aquí para explorar cómo RapidMiner puede transformar tus proyectos de ciencia de datos.

Rúbrica de Evaluación UNIDAD IV

Práctica: Conexión de Apache Superset a múltiples fuentes de datos
Puntaje total: 40 puntos
Escala: Excelente (4) · Bueno (3) · Básico (2) · Insuficiente (1)
ALUMNO: ROMÁN OMAR FISCAL POLITO
CALIFICACIÓN OBTENIDA: 40%

1. Conexión a base de datos SQL (20 puntos)

Nivel	Descripción
4 – Excelente (20 pts)	La conexión SQL es correcta, funcional y documentada. El dataset se crea sin errores y se valida mediante consulta o visualización.
3 – Bueno (16 pts)	La conexión funciona, pero la documentación o validación es parcial.
2 – Básico (12 pts)	La conexión presenta errores menores o depende de apoyo externo.
1 – Insuficiente (8 pts)	No logra establecer la conexión o no crea el dataset.

2. Conexión a archivos CSV (20 puntos)

Nivel	Descripción
4 – Excelente (20 pts)	El CSV se importa correctamente, con tipos de datos bien definidos y visualización funcional.
3 – Bueno (16 pts)	El CSV se importa, pero con ajustes incompletos o visualización básica.
2 – Básico (12 pts)	Importación incompleta o con errores de tipos de datos.
1 – Insuficiente (8 pts)	No logra importar el archivo o el dataset no funciona.

Alumno: Román Omar Fiscal Polito

Práctica de Laboratorio-----UNIDAD IV
Conexión de Apache Superset a múltiples fuentes de datos
Objetivo
Configurar y validar conexiones en Apache Superset hacia **bases de datos SQL**, **archivos CSV** y **otras fuentes compatibles**

1.- Conexión a una base de datos SQL

Conectar Superset a una base de datos PostgreSQL y crear un dataset.

Paso 1. Ingresar a Superset
Acceder vía navegador:
http://localhost:8088

Paso 2. Agregar base de datos
Ruta:
Settings → Database Connections → + Database
Seleccionar **PostgreSQL**.

Paso 3. Configurar la cadena de conexión
Ejemplo:
postgresql://usuario:password@localhost:5432/ventas_db
Campos:

- Host: localhost
- Puerto: 5432
- Base de datos: ventas_db
- Usuario: usuario
- Contraseña: password

Paso 4. Probar conexión
Clic en **Test Connection**
Mensaje esperado: *Connection looks good!*

Paso 5. Crear Dataset
Ruta:
Data → Datasets → + Dataset
Seleccionar:

- Base de datos: ventas_db
- Esquema: public
- Tabla: ventas

Evidencia esperada

- Dataset ventas disponible en Superset.

2. Conexión a archivos CSV

Importar un archivo CSV y crear una visualización básica.

Paso 1. Subir archivo CSV
Ruta:
Data → Upload a CSV
Ejemplo de archivo:
ventas_mensuales.csv

Paso 2. Configuración

- Nombre del Dataset: ventas_csv
- Revisar tipos de datos:
  - fecha → DATE
  - monto → FLOAT
  - region → STRING

Paso 3. Confirmar carga
Mensaje esperado: *CSV uploaded successfully*

Paso 4. Crear visualización
Ruta:
Charts → + Chart

- Dataset: ventas_csv
- Tipo: Bar Chart
- Métrica: SUM(monto)
- Dimensión: region

Evidencia esperada

- Gráfica de ventas por región.

3. Otras fuentes compatibles

Identificar otra fuente compatible y explicar cómo conectarla. **Google Sheets**

Opción técnica:

1. Crear una base intermedia (PostgreSQL).
2. Importar datos desde Google Sheets usando ETL (Python).
3. Conectar Superset a PostgreSQL (ya configurado).

Alternativa:

- BigQuery
- Amazon RDS
- SQLite
- Microsoft SQL Server

Rúbrica de Evaluación Examen Práctico valor 50%

Práctica: Procesamiento de datos con MapReduce
Puntaje máximo: 50 puntos
Escala por criterio: Excelente · Bueno · Básico · Insuficiente

ALUMNO: Román Omar Fiscal Polito

1. Diseño del modelo MapReduce (10 puntos)

Nivel	Descripción	Puntos
Excelente	Define correctamente las fases Map y Reduce, con claridad en entradas y salidas.	10
Bueno	Define Map y Reduce con ligeras imprecisiones conceptuales.	8
Básico	Identificación parcial de las fases o confusión menor.	6
Insuficiente	No distingue adecuadamente Map y Reduce.	4

2. Implementación del Mapper (10 puntos)

Nivel	Descripción	Puntos
Excelente	Mapper funcional, correcto manejo de entrada y salida (clave–valor).	10
Bueno	Mapper funcional con detalles menores de formato o eficiencia.	8
Básico	Mapper incompleto o con errores menores.	6
Insuficiente	Mapper no funcional o ausente.	4

3. Implementación del Reducer (10 puntos)

Nivel	Descripción	Puntos
Excelente	Reducer correcto, agrega datos adecuadamente y produce resultados precisos.	10
Bueno	Reducer funcional con ligeros errores de lógica o formato.	8
Básico	Reducer incompleto o con errores frecuentes.	6
Insuficiente	Reducer no funcional o ausente.	4

4. Ejecución y resultados obtenidos (10 puntos)

Nivel	Descripción	Puntos
Excelente	Job ejecuta correctamente y produce resultados esperados.	10
Bueno	Job ejecuta con resultados parciales o ajustes menores.	8
Básico	Job ejecuta con errores que afectan resultados.	6
Insuficiente	Job no ejecuta o no genera salida válida.	4

5. Análisis y reflexión técnica (10 puntos)

Nivel	Descripción	Puntos
Excelente	Explica claramente el funcionamiento de MapReduce y justifica decisiones.	10
Bueno	Análisis adecuado pero poco profundo.	8
Básico	Respuestas descriptivas sin análisis crítico.	6
Insuficiente	Análisis incorrecto o incompleto.	4

EXAMEN Práctico 50%

Procesamiento de datos con MapReduce

Objetivo de aprendizaje

Aplicar el paradigma **MapReduce** para procesar grandes volúmenes de datos, comprendiendo la separación de responsabilidades entre las fases **Map** y **Reduce** y su utilidad en entornos de Big Data.

Problema:

Una empresa desea **analizar registros de ventas** almacenados en un archivo de texto distribuido para conocer **el total de ventas por producto**.

Datos de entrada

Archivo: ventas.txt
Laptop,12000
Mouse,300
Laptop,12000
Teclado,800
Mouse,300
Laptop,12000
Formato:
Producto,Monto

Actividad a realizar

Desarrollar un programa **MapReduce** que:

- Lea el archivo de ventas.
- Agrupe las ventas por producto.
- Calcule el **total de ventas por producto**.

DESARROLLO DE LA PRÁCTICA

A) – Diseño conceptual

1. Fase Map

- Entrada: Línea de texto
- Salida: (producto, monto)

2. Fase Reduce

- Entrada: (producto, [montos])
- Salida: (producto, total)

B) – Implementación (Python con Hadoop Streaming)

1. Mapper (mapper.py)

```
#!/usr/bin/env python3
import sys

for line in sys.stdin:
    line = line.strip()
    producto, monto = line.split(",")
```

```
    print(f"{producto}\t{monto}")
```

2. Reducer (reducer.py)

```
#!/usr/bin/env python3
import sys

ventas_actuales = {}
for line in sys.stdin:
    producto, monto = line.strip().split("\t")
    ventas_actuales[producto] = ventas_actuales.get(producto, 0) + float(monto)

for producto, total in ventas_actuales.items():
    print(f"{producto}\t{total}")
```

Ejecución en Hadoop

1. Subir archivo a HDFS

```
hdfs dfs -mkdir /ventas
hdfs dfs -put ventas.txt /ventas
```

2. Ejecutar el job MapReduce

```
hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar \
-input /ventas/ventas.txt \
-output /ventas/salida \
-mapper mapper.py \
-reducer reducer.py
```

3. Ver resultados

```
hdfs dfs -cat /ventas/salida/part-00000
```

Salida esperada

```
Laptop    36000
Mouse     600
Teclado   800
```