Curso: Inteligencia Artificial
Profesor: MSc. Steven Pacheco Portuguez
Semestre: II 2024
Valor: 20%
Fecha de entrega: 12 de septiembre 2024

Proyecto I

Este proyecto tiene como objetivo principal aplicar diversas técnicas de clasificación de datos aplicados para dos conjuntos de datos, esto permite explorar diversas herramientas relacionadas al Machine Learning, y contribuir al desarrollo del conocimiento a partir de la investigación.

El primer conjunto de datos corresponde al Instituto Nacional de Diabetes y Enfermedades Digestivas y Renales y el objetivo de este conjunto de datos es predecir de forma diagnóstica si un paciente tiene o no tiene diabetes, dado un conjunto de mediciones realizadas al paciente.

Link: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

El segundo conjunto de datos es a escoger, seleccione un conjunto de datos de su interés que permita hacer una clasificación binaria, se recomienda que utilice un conjunto de datos tabular, y no hacer uso de conjuntos de datos para computer visión o NLP (Natural Language Processing). Mencione en su informe porque seleccionaron este conjunto de datos.

Los siguientes pasos deben de realizarse ambos conjuntos de datos:

Exploración y Preprocesamiento de Datos:

- Explore el conjunto de datos para comprender la naturaleza de las características (features) del conjunto de datos.
- Maneje valores faltantes, datos sobresalientes en caso de ser requerido y realice una exploración estadística básica para observar el comportamiento de los datos. Justifique sus observaciones
- Evalue si su conjunto de datos se encuentra balanceado, puede utilizar una visualización estadística da la distribución de las clases. Se recomienda usar matplotlib o seaborn
- Separe el conjunto de datos en entrenamiento 70% para datos de entrenamiento, 15% para validación y 15% para testing.
- Utilice el set de datos de validación para verificar la convergencia de su modelo.

Modelos:

Debe utilizar regresión logística, KNN, puede utilizar la biblioteca de su preferencia como Scikit-learn, realice múltiples ejecuciones de sus modelos cambiando valores de hiperparametros que le permitan hallar el modelo que presenta mejores resultados.

Evalue los modelos utilizando métricas (mínimo Accuracy, Precision, Recall) basado en el conjunto de datos de prueba seleccionado, adicionalmente incluya la matriz de confusión del modelo final.

A lo largo de los experimentos los conjuntos de entrenamiento y pruebas no pueden variar para asegurar igualdad de condiciones durante los experimentos.

Nota: En este proyecto no se le solicita que desarrolle manualmente el código para estos modelos.

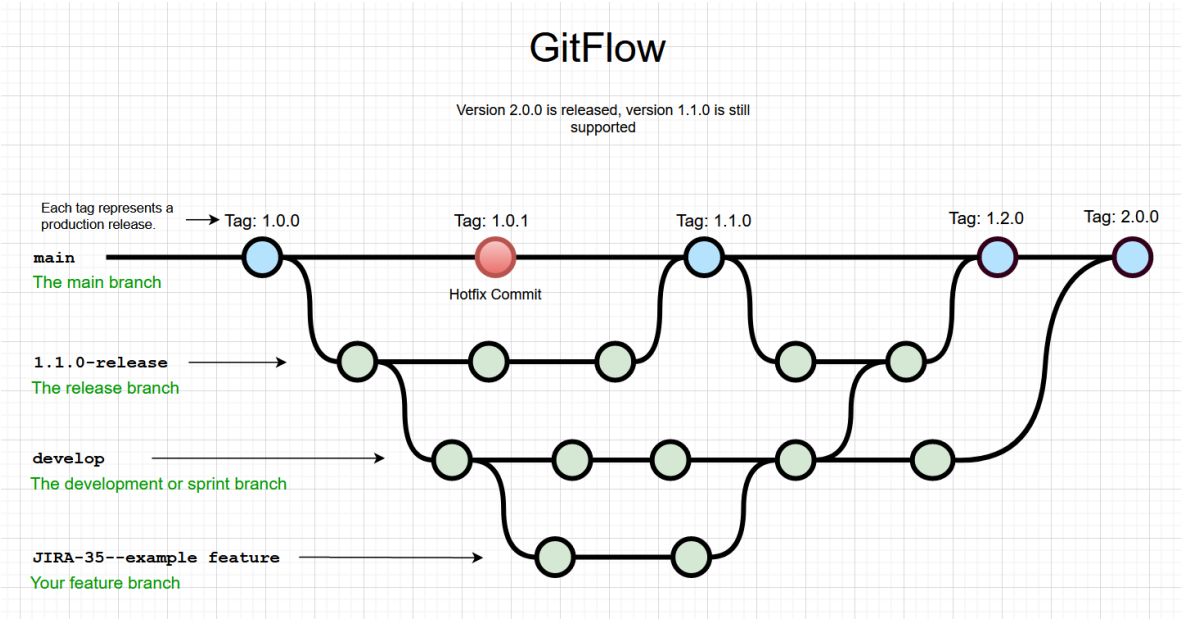
Entregables del proyecto:

- Jupyter notebook con la exploración de los datos y sus visualizaciones, además de incluir el resultado de aplicación de los modelos, así como la evaluación de métricas para cada. Debe incluir los resultados.
- Informe: Cree un informe en formato de artículo científico (Latex) donde describa los experimentos realizados y realice una comparación de los resultados de cada modelo para cada conjunto de datos.

  La estructura del informe básica es la siguiente: Abstract, Introducción, Metodología, Resultados, Discusión y Conclusiones, Bibliografía. Debe entregar el código fuente y el PDF.

Conjunto de datos de Diabetes		
Criterios	Puntuación máxima	Puntuación obtenida
Análisis del conjunto de datos y features	5	
Análisis de regresión logística	15	
Análisis de KNN	15	
Comparación de modelos	10	
Conjunto de datos seleccionado		
Criterios	Puntuación máxima	Puntuación obtenida
Análisis del conjunto de datos y features	5	
Análisis de regresión logística	15	
Análisis de KNN	15	
Comparación de modelos	10	
Aspectos Generales		
Criterios	Puntuación máxima	Puntuación obtenida
Completitud de entregables	5	
Estructura de artículo científico	5	
Aspectos Extra (SRE)		
Criterio	Puntuación máxima	Puntuación obtenida
Usar Gitflow como proceso de colaboración y utilizar tags de versionamiento (main branch). Deben participar los 2 integrantes.	5	
Total	105	

Anexo:


GitFlow
Version 2.0.0 is released, version 1.1.0 is still supported
Each tag represents a production release.
Tag: 1.0.0
Tag: 1.0.1
Tag: 1.1.0
Tag: 1.2.0
Tag: 2.0.0
main
The main branch
Hotfix Commit
1.1.0-release
The release branch
develop
The development or sprint branch
JIRA-35--example feature
Your feature branch