
RESULTADO DE APRENDIZAJE

RdA de la asignatura:

- Plantear los conceptos fundamentales del aprendizaje automático, incluyendo los principios básicos, técnicas de preprocesado de datos, métodos de evaluación y ajuste de modelos, destacando su importancia en el análisis y resolución de problemas de datos.

RdA de la actividad:

- Comprender la importancia de la selección y extracción de atributos en la construcción de modelos eficientes.
- Implementar métodos básicos como PCA y SVD para reducir dimensionalidad.
- Analizar aplicaciones de factorización de matrices no negativas en aprendizaje automático.

INTRODUCCIÓN

Pregunta inicial: ¿Por qué es importante reducir la dimensionalidad de un conjunto de datos al construir un modelo de aprendizaje automático?

DESARROLLO

Actividad 1: Selección de Atributos

Se preparará un modelo GPT interactivo con el cual los estudiantes podrán explorar conceptos clave, ventajas y desventajas de técnicas de selección de atributos.

¿Cómo lo haremos?

- **Clase interactiva:** Los estudiantes interactuarán con un modelo GPT diseñado para explicar las técnicas de selección de atributos y sus aplicaciones prácticas.

Enlace al GPT: Tutor AA - Selección de Atributos

Verificación de aprendizaje: Los estudiantes responderán las preguntas del GPT proporcionado.

Actividad 2: Análisis de Componentes Principales (PCA)

Se utilizará un modelo GPT para repasar conceptos clave de PCA y guiar a los estudiantes en su implementación práctica.

¿Cómo lo haremos?

- **Clase interactiva:** Los estudiantes interactuarán con un modelo GPT diseñado para sus conocimientos en PCA.

Enlace al GPT: Evaluador AA - PCA

- **Ejemplo práctico en Python:**

```
from sklearn.decomposition import PCA
from sklearn.datasets import load_iris
import pandas as pd

# Cargar conjunto de datos Iris
data = load_iris()
X = pd.DataFrame(data.data, columns=data.feature_names)
display(X.head())

# Reducir a 2 componentes principales
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)

# Proporción de varianza explicada
explained_variance_ratio = pca.explained_variance_ratio_
print("Varianza explicada por cada componente principal:")
print(explained_variance_ratio)

# Matriz de componentes principales
print("\nMatriz de componentes principales:")
print(pca.components_)

# Datos transformados
df_pca = pd.DataFrame(X_pca, columns=["PC1", "PC2"])
print("\nDatos transformados:")
display(df_pca.head())
```

Verificación de aprendizaje: Los estudiantes responderán las preguntas del GPT proporcionado.

Actividad 3: Descomposición en Valores Singulares (SVD)

Clase magistral para explicar los fundamentos de SVD, seguida de un ejemplo práctico de implementación en Python.

¿Cómo lo haremos?

- **Clase magistral:** Introducción matemática a SVD y sus aplicaciones en reducción de dimensionalidad y sistemas de recomendación.
- **Ejemplo práctico en Python:**

```
import numpy as np

# Matriz de ejemplo
A = np.array([[1, 2], [3, 4], [5, 6]])
U, S, VT = np.linalg.svd(A)

print("Matriz Original:\n", A)
print("Matriz U:\n", U)
print("Valores Singulares:\n", S)
print("Matriz V Transpuesta:\n", VT)
```

- **Discusión guiada:** Los estudiantes analizarán el significado de cada matriz resultante (U, S, VT) y cómo estas se relacionan con la matriz original.

Verificación de aprendizaje: Los estudiantes responderán cómo se puede usar SVD para compresión de datos y reducción de ruido.

Actividad 4: Factorización de Matrices No Negativas

Lectura asignada sobre aplicaciones prácticas de factorización de matrices no negativas (NMF).

¿Cómo lo haremos?

- **Lectura asignada:** Artículo “Non-negative Matrix Factorization: A Survey”.
- **Análisis:** Leer el resumen y la introducción del artículo para discutirlo la siguiente clase.

CIERRE

Tarea: Culminar la Actividad 4.

Pregunta de investigación:

1. ¿Puedo hacer PCA para datos categóricos?
2. ¿Qué sistemas de recomendación usan SVD?