

ACTA DE REUNIÓN

Fecha: 02 abril de 2025

Hora de inicio: 7:00 a.m.

Hora de finalización: 8:00 a.m.

Lugar: Sala de juntas de la escuela de ingeniería de sistemas de la UIS

Asistentes:

Daniel Alejandro Sánchez Rodríguez (Estudiante)

Jorge Eduardo Suárez Cortés (Estudiante)

César Augusto Aceros Moreno (Profesor)

Eliana Martha Bonalde Marcano (Profesora)

Andrés Leonardo González Gómez (Profesor)

David Edmundo Romo Bucheli (Profesor)

Sonia Cristina Gamboa Sarmiento (Profesora)

Orden del Día

1. Definición del tema principal del proyecto y su desarrollo.
2. Análisis de los lenguajes de programación y sus características.
3. Desinterés en el uso de cuadernos virtuales y preferencia por herramientas de inteligencia artificial.
4. Desarrollo de funcionalidades adicionales no incluidas en la herramienta (interfaz visual, gráficos, etc.).
5. Organización del contenido en la plataforma Moodle.
6. Comparación de herramientas disponibles como Python, MATLAB y R.
7. Desarrollo de herramientas que permitan el uso de diferentes lenguajes de programación.

Desarrollo de la Reunión

1. **Definición del tema del proyecto:**
 - Se discutió la idea principal del proyecto y cómo se desarrollará en sus diferentes etapas, acordando que el enfoque debe ser mejorar la enseñanza y el aprendizaje mediante el uso de herramientas digitales.

2. Análisis de los lenguajes de programación:

- Se resaltó la importancia de las características generales de Python, MATLAB y R, centrándose en sus usos y herramientas disponibles. Donde se mencionaron las plataformas que facilitan su manejo, como lo son MATLAB Grader y R Notebooks. Posteriormente no se eligió un lenguaje específico, dejando abierta la decisión para más adelante de acuerdo a una serie de posibilidades que se manejan mediante un cuadro comparativo.

3. Desinterés en el uso de cuadernos virtuales y preferencia por herramientas de inteligencia artificial:

- Se ha observado que los estudiantes están perdiendo el interés en utilizar cuadernos virtuales (como notebooks o plataformas tipo Grader) debido a la facilidad que ofrecen las herramientas de inteligencia artificial para desarrollar ejercicios y trabajos. Esta tendencia ha generado una disminución en el compromiso del estudiante con su propio proceso de aprendizaje, ya que muchos recurren directamente a la IA sin dedicar tiempo a comprender los conceptos fundamentales. Ante esto, se hace necesario diseñar estrategias pedagógicas que permitan integrar adecuadamente la tecnología, promoviendo el pensamiento crítico y la comprensión.

4. Desarrollo de funcionalidades adicionales:

- Se rescató la importancia de incluir gráficos e interfaces visuales en las herramientas que no tengan estas opciones, ya que con la ayuda de estos elementos gráficos se facilita la comprensión de los resultados independientemente del lenguaje de la herramienta y así tener una retroalimentación adecuada en el estudiante.

5. Organización de unidades en Moodle:

- Se tendrá en cuenta en analizar esta idea dejando establecido la parte pedagógica dentro de la cuales se tienen entendido trabajar, para construir un entorno virtual adecuado para el curso.

6. Comparación entre Python, MATLAB y R:

- Se indicó la elaboración de un cuadro comparativo sobre las ventajas y desventajas de cada herramienta, asignando a los estudiantes la tarea de realizar este análisis tomando en cuenta los aspectos más importantes de cada una y posteriormente socializar esta con los profesores de estadística inferencial.

7. Desarrollo de herramientas que permitan el uso de distintos lenguajes:

- Se propuso la creación de herramientas flexibles que no dependieran de un solo lenguaje de programación, es decir que las herramientas a elaborar se tendrán que dejar de manera agnóstica para que el usuario tenga la decisión del libre albedrío o el lenguaje del gusto.

Desarrollo del Problema

Se estableció que se trabajará sobre un conjunto de temas en diferentes unidades, lo que permitirá hacer pruebas más efectivas y evaluar el funcionamiento del proyecto. La distribución por unidades será la siguiente:

- **Primera unidad:** Temas 1.3 y 1.4.
- **Segunda unidad:** Temas 2.3 y 2.4.
- **Tercera unidad:** Temas 3.3 y 3.4.

Tabla comparación de Python, Matlab y R.

Característica	Python	MATLAB	R
Facilidad de uso	Sintaxis sencilla y clara, fácil de aprender.	Sintaxis estructurada y más formal.	Requiere conocimientos de estadística para aprovecharlo al máximo.
Bibliotecas para el proyecto	NumPy, Pandas, SciPy, Matplotlib.	Integrado con funciones matemáticas avanzadas y toolboxes especializadas.	ggplot2, dplyr, tidyr, Shiny.
Entorno de desarrollo	Jupyter Notebook, Google Colab, PyCharm, vs. Code.	MATLAB IDE.	RStudio, R Notebooks.
Herramientas educativas	Google Colab, Jupyter Notebook.	MATLAB Grader.	R Notebooks, R Markdown.
Visualización de datos	Matplotlib, Seaborn, Plotly.	Gráficos avanzados dentro del entorno MATLAB.	ggplot2, base R graphics.

Uso en la nube	Google Colab, AWS Lambda, Azure ML.	MATLAB Online, integración con AWS y Azure.	RStudio Cloud.
Costo	Open-source (gratuito).	Requiere licencia de pago. (Licencia de la Universidad)	Open-source (gratuito).
Rendimiento	Rápido con optimización adecuada, pero depende de las bibliotecas.	Optimizado para cálculos numéricos, excelente desempeño.	Rápido en análisis estadísticos, pero más lento en cálculos generales.
Calificación de ejercicios	Evaluaciones automáticas posibles con nbgrader en Jupyter Notebook y bibliotecas como pytest, doctest, o Otter-Grader para notebooks interactivos.	MATLAB Grader, plataforma específica para la evaluación de ejercicios de programación, permite autocalificación y retroalimentación automatizada.	Evaluaciones con learnr (para cuestionarios interactivos en R Markdown) o paquetes como testthat y gradethis para calificar código automáticamente.
Áreas de uso	Ciencia de datos, inteligencia artificial, desarrollo web, automatización, análisis financiero, computación científica	Procesamiento de señales, simulación numérica, control, ingeniería, robótica	Estadística, análisis de datos, visualización, bioestadística, ciencia social cuantitativa
Ventajas	<ul style="list-style-type: none"> - Amplias librerías (SciPy, statsmodels). - Fácil integración con machine learning. - Código más versátil (no sólo estadística). 	<ul style="list-style-type: none"> - Funciones estadísticas robustas. - Excelente para análisis numérico. - Muy buenas herramientas gráficas integradas. 	<ul style="list-style-type: none"> - Diseñado específicamente para análisis estadístico. - Gran cantidad de paquetes estadísticos (como infer, boot, lmtest). - Comunidad muy enfocada en

			estadística y ciencia de datos.
Desventajas	<ul style="list-style-type: none"> - A veces requiere más configuración para análisis complejos. - Algunas librerías estadísticas profundas no son tan maduras como en R. 	<ul style="list-style-type: none"> - Licencia costosa. - Menor flexibilidad en machine learning comparado con Python. - No tan "natural" para big data moderno. 	<ul style="list-style-type: none"> - Curva de aprendizaje empinada si no vienes de estadística. - Sintaxis menos intuitiva para usuarios nuevos. - Rendimiento bajo en tareas fuera del análisis estadístico.