DIPLOMADO EN

DATA SCIENCE

CIENCIA DE DATOS — FORMACIÓN PROFESIONAL

12 Niveles	240 Horas Académicas	36 Prácticas	Edición 2026

■ Matemáticas y Estadística	■ Programación y Herramientas	■ Machine Learning e Inteligencia Artificial	■ Análisis de Datos y Toma de Decisiones

Disciplina científica e interdisciplinaria que combina matemáticas, estadística, programación e inteligencia artificial para extraer conocimiento de valor y patrones ocultos a partir de grandes volúmenes de datos.

www.uneweb.edu.ve

Descripción General del Diplomado

Este Diplomado en Data Science forma profesionales capaces de gestionar el ciclo completo del dato: desde la recopilación y limpieza hasta el modelado predictivo, el despliegue en producción y la comunicación ejecutiva de resultados. Con 12 niveles progresivos y 240 horas académicas, el participante desarrolla competencias en matemáticas, estadística, Python, SQL, Machine Learning, Deep Learning, MLOps y herramientas de BI, siguiendo los estándares de la industria 2026.

Modalidad	Teórico-Práctica (40% teoría / 60% práctica)
Duración total	240 horas académicas (12 niveles × 20 horas)
Prácticas	3 prácticas por nivel (36 prácticas totales)
Certificación	Diplomado en Data Science con acreditación de horas — UneWeb 2026
Herramientas	Python, R, SQL, Jupyter, Pandas, NumPy, Scikit-learn, TensorFlow, PyTorch, Spark, Tableau, Power BI, BigQuery, dbt, Airflow, MLflow, Docker, AWS/GCP
Nivel de salida	Data Scientist Junior-Senior, ML Engineer, Data Analyst, BI Developer, MLOps Engineer

Requisitos de Ingreso

El Diplomado está estructurado en 3 fases de 4 niveles cada una. Los requisitos varían según el punto de entrada:

Fase 1 — Niveles 1–4 Fundamentos	Fase 2 — Niveles 5–8 ML e IA Aplicada	Fase 3 — Niveles 9–12 Avanzado y Producción
• Bachillerato o equivalente • Manejo básico de computadora y hojas de cálculo • Lógica matemática básica (álgebra) • Inglés técnico básico (lectura)	• Completar Fase 1 o demostrar equivalencia • Python intermedio (Pandas, NumPy) • SQL intermedio • Estadística descriptiva e inferencial	• Completar Fase 2 o demostrar equivalencia • ML supervisado y no supervisado dominado • Python avanzado y manejo de APIs • Experiencia en al menos 1 proyecto real de datos

Estructura Curricular — 12 Niveles

Niv.	Fase	Título del Nivel	Horas	Prácticas
1	I	Fundamentos de Programación con Python	20 h	3
2	I	Matemáticas y Estadística para Data Science	20 h	3
3	I	SQL y Bases de Datos para Ciencia de Datos	20 h	3
4	I	Análisis Exploratorio de Datos (EDA) y Visualización	20 h	3
5	II	Machine Learning Supervisado	20 h	3
6	II	Machine Learning No Supervisado y Feature Engineering	20 h	3
7	II	Deep Learning y Redes Neuronales	20 h	3
8	II	NLP: Procesamiento del Lenguaje Natural	20 h	3
9	III	MLOps: Despliegue y Ciclo de Vida de Modelos	20 h	3

10	III	Big Data: Procesamiento a Gran Escala	20 h	3
11	III	IA Generativa y LLMs Aplicados a Data Science	20 h	3
12	III	Proyecto Final: Data Science en Producción	20 h	3
		TOTAL DEL DIPLOMADO	**240 h**	**36**

NIVEL 1

20 Horas | 3 Prácticas

Fundamentos de Programación con Python

Objetivo: Dominar la sintaxis de Python y las librerías esenciales para el manejo de datos, estableciendo la base programática de todo el diplomado.

■ Programación / Herramientas	■ Matemáticas / Estadística
• Variables, tipos de datos, estructuras de control y funciones • Listas, diccionarios, conjuntos y tuplas • POO básica: clases, objetos, métodos e herencia • Manejo de archivos: CSV, JSON, TXT	• NumPy: arrays multidimensionales, operaciones vectorizadas • Álgebra básica con arrays: suma, multiplicación, transposición • Pandas: Series y DataFrames — creación, indexación y slicing

Módulo 1.1 — Python para Ciencia de Datos (10h)

- Entorno de trabajo: Anaconda, Jupyter Notebook, VS Code y Google Colab
- Tipos de datos: int, float, str, bool, None — conversiones y operaciones
- Estructuras de control: if/elif/else, for, while, list comprehensions
- Funciones: definición, parámetros, *args, **kwargs, lambda y decoradores
- Manejo de errores: try/except/finally y excepciones personalizadas
- Módulos y paquetes: pip, importaciones y creación de módulos propios

Módulo 1.2 — NumPy y Pandas Fundamentos (10h)

- NumPy: creación de arrays, indexado booleano, broadcasting y ufuncs
- Pandas: lectura de CSV/Excel, describe(), info(), value_counts()
- Filtrado, ordenado, groupby y merge de DataFrames
- Tratamiento de valores nulos: isnull(), dropna(), fillna()
- Operaciones con fechas y strings en Pandas
- Exportar resultados: to_csv(), to_excel(), to_json()

PRÁCTICAS DEL NIVEL

P1 Análisis de dataset de ventas con Pandas: limpieza, agrupación y resumen estadístico en Jupyter Notebook

P2 Calculadora de estadísticas con NumPy: media, mediana, desv. estándar y percentiles sobre datos reales

P3 Pipeline de limpieza de datos: cargar CSV sucio → detectar nulos → imputar → exportar dataset limpio

NIVEL 2 20 Horas | 3 Prácticas

Matemáticas y Estadística para Data Science

Objetivo: Comprender los fundamentos matemáticos y estadísticos que sustentan los algoritmos de Machine Learning y el análisis de datos sin requerir conocimientos previos avanzados.

■ Matemáticas / Estadística	■ Programación / Herramientas
• Álgebra lineal: vectores, matrices, multiplicación y determinantes • Cálculo: derivadas, gradientes y regla de la cadena (intuitivo) • Probabilidad: espacio muestral, eventos, P(A∩B), P(A|B), Bayes	• SciPy: distribuciones estadísticas, tests de hipótesis • Matplotlib/Seaborn: histogramas, boxplots y heatmaps estadísticos • Implementar distribuciones y tests con Python desde cero

Módulo 2.1 — Álgebra Lineal y Cálculo Aplicado (8h)

- Vectores: norma, producto punto, similitud coseno — aplicación en recomendadores
- Matrices: operaciones, transpuesta, inversa y descomposición (SVD, PCA intuición)
- Gradiente: qué es, para qué sirve en ML — descenso del gradiente paso a paso
- Autovalores y autovectores: intuición geométrica y su rol en PCA
- Función de pérdida: MSE, MAE, Cross-Entropy — por qué las minimizamos

Módulo 2.2 — Estadística para ML (12h)

- Estadística descriptiva: media, mediana, moda, varianza, desviación estándar, curtosis
- Distribuciones: Normal, Binomial, Poisson, Bernoulli — parámetros y aplicaciones
- Inferencia estadística: estimación puntual e intervalo de confianza
- Tests de hipótesis: H0 y H1, p-value, t-test, chi-cuadrado, ANOVA
- Correlación vs. causalidad: Pearson, Spearman, covarianza
- Regresión lineal simple desde la estadística: MCO, R^2, p-value del coeficiente

PRÁCTICAS DEL NIVEL

P1 Implementar descenso del gradiente desde cero con NumPy para regresión lineal — visualizar convergencia

P2 Análisis estadístico completo de un dataset: distribuciones, outliers, correlaciones y tests de hipótesis

P3 Estudio A/B test: diseñar experimento, calcular tamaño muestral, ejecutar t-test y concluir con p-value

NIVEL 3 20 Horas | 3 Prácticas

SQL y Bases de Datos para Ciencia de Datos

Objetivo: Dominar SQL como herramienta central de extracción y transformación de datos, conectar bases de datos con Python y trabajar con BigQuery para análisis a escala.

■ Programación / Herramientas	■ Matemáticas / Estadística
• PostgreSQL y MySQL: instalación, DBeaver y SQLiteOnline • Python + SQL: psycopg2, SQLAlchemy y pandas.read_sql() • BigQuery: consultas en la nube sobre datasets públicos masivos	• Álgebra relacional: proyección, selección, join y unión • Modelado ER: diseño de esquemas para proyectos de datos • Normalización y desnormalización: trade-offs para analytics

Módulo 3.1 — SQL para Análisis de Datos (10h)

- SELECT avanzado: DISTINCT, aliases, CASE WHEN, COALESCE, CAST
- JOINs: INNER, LEFT, RIGHT, FULL OUTER, SELF JOIN y antipatrones
- Agregaciones: COUNT, SUM, AVG, MIN, MAX con GROUP BY y HAVING
- CTEs (WITH): legibilidad, reutilización y CTEs recursivas
- Window Functions: ROW_NUMBER, RANK, LAG, LEAD, SUM OVER PARTITION BY
- Optimización: índices, EXPLAIN ANALYZE y query tuning para grandes tablas

Módulo 3.2 — Bases de Datos y Python (10h)

- Diseño de base de datos para proyectos de Data Science: entidades, relaciones y claves
- pandas + SQL: read_sql_query(), to_sql() — puente entre bases de datos y DataFrames
- SQLAlchemy ORM básico: modelos, sesiones y consultas con Python
- BigQuery: crear proyecto, ejecutar consultas sobre TB de datos, costes y optimización
- dbt básico: modelos SQL versionados, tests de calidad y documentación automática
- NoSQL introductorio: MongoDB, Redis y casos de uso vs. SQL relacional

PRÁCTICAS DEL NIVEL

P1 Análisis de ventas con SQL puro: JOINs, CTEs y window functions sobre base de datos Northwind

P2 Pipeline Python+SQL: leer datos de PostgreSQL con pandas, transformar y cargar de vuelta (ETL básico)

P3 Análisis de 10M+ registros en BigQuery: consultas optimizadas, CTEs y exportación a DataFrame

NIVEL 4 20 Horas | 3 Prácticas

Análisis Exploratorio de Datos (EDA) y Visualización

Objetivo: Desarrollar la capacidad de explorar, entender y comunicar visualmente cualquier dataset antes de modelar, usando Matplotlib, Seaborn, Plotly y Tableau.

■ Matemáticas / Estadística	■ Análisis / Negocio
• Distribuciones y outliers: IQR, Z-score, detección y tratamiento • Correlaciones: Pearson, Spearman, heatmaps y pairplots • Sesgos estadísticos y su impacto en los modelos	• Storytelling con datos: estructura narrativa y audiencia • Tableau/Power BI: dashboards ejecutivos desde datos crudos • Principios de diseño visual: color, contraste y jerarquía

Módulo 4.1 — EDA Profundo con Python (10h)

- Perfil de datos: dtypes, nulos, duplicados, cardinalidad y distribuciones
- Matplotlib: anatomía de una figura, subplots, estilos y exportación
- Seaborn: histplot, boxplot, violinplot, heatmap, pairplot y FacetGrid
- Plotly Express e Interactive: gráficos interactivos, zoom y hover
- Detección de outliers: IQR, Z-score, Isolation Forest visual
- Análisis multivariado: scatter matrix, correlaciones y PCA visual

Módulo 4.2 — Visualización y Comunicación de Datos (10h)

- Principios de diseño de datos: elección del gráfico correcto según el mensaje
- Tableau Desktop: conexión a datos, dimensiones, métricas y dashboards
- Power BI: Power Query, DAX básico, visuales y publicación en la nube
- Storytelling con datos: estructura del insight (dato → insight → acción)
- Plotly Dash: crear un dashboard web interactivo con Python en 2 horas
- Looker Studio: conectar BigQuery y crear dashboard ejecutivo público

PRÁCTICAS DEL NIVEL

P1 EDA completo de un dataset de e-commerce: perfil, distribuciones, outliers, correlaciones y 15 visualizaciones

P2 Dashboard interactivo con Plotly Dash: KPIs en tiempo real, filtros y drill-down por categoría

P3 Presentación ejecutiva de 10 slides con insights del EDA — storytelling con datos para audiencia no técnica

NIVEL 5 20 Horas | 3 Prácticas

Machine Learning Supervisado

Objetivo: Implementar, evaluar y optimizar los principales algoritmos de ML supervisado para clasificación y regresión usando Scikit-learn sobre problemas empresariales reales.

■ ML / IA	■ Matemáticas / Estadística
• Regresión: Lineal, Ridge, Lasso, ElasticNet, Polinomial • Clasificación: Logística, KNN, SVM, Árbol de Decisión, Random Forest, XGBoost • Evaluación: RMSE, R^2, Accuracy, Precision, Recall, F1, ROC-AUC, Matriz de confusión	• Regularización: overfitting, bias-variance tradeoff, curvas de aprendizaje • Validación cruzada: K-Fold, Stratified, Leave-One-Out • Optimización de hiperparámetros: GridSearchCV, RandomizedSearch, Bayesian

Módulo 5.1 — Regresión y Clasificación (10h)

- Pipeline de ML: datos → preprocesado → modelo → evaluación → ajuste
- Regresión Lineal Multiple: interpretación de coeficientes, multicolinealidad y VIF
- Regresión Logística: odds ratio, umbral de clasificación y curva ROC
- Árboles de Decisión: criterio de división, profundidad y poda
- Ensemble methods: Bagging (Random Forest), Boosting (XGBoost, LightGBM, CatBoost)
- Support Vector Machines: kernel trick, margen y casos de uso óptimos

Módulo 5.2 — Evaluación, Tuning y Pipelines (10h)

- Scikit-learn Pipeline: encadenar preprocesado y modelo en un objeto reproducible
- ColumnTransformer: transformaciones diferentes por tipo de columna
- Manejo de desbalance de clases: SMOTE, undersampling, class_weight
- Feature importance: Permutation Importance, SHAP values para interpretabilidad
- Optuna y Hyperopt: optimización bayesiana de hiperparámetros
- Experimento reproducible: random_state, joblib y versionado de modelos

PRÁCTICAS DEL NIVEL

P1 Predicción de churn de clientes: comparar 5 algoritmos, optimizar XGBoost con Optuna e interpretar con SHAP

P2 Predicción de precios de vivienda: regresión con Lasso/Ridge, ingeniería de features y validación cruzada

P3 Pipeline completo de clasificación de crédito: preprocesado, modelo, umbral óptimo e informe de métricas

NIVEL 6 20 Horas | 3 Prácticas

ML No Supervisado y Feature Engineering

Objetivo: Dominar algoritmos de clustering, reducción de dimensionalidad y la creación sistemática de features que maximizan el rendimiento de los modelos.

■ ML / IA	■ Matemáticas / Estadística
• Clustering: K-Means, DBSCAN, Hierarchical, GMM — selección de K y validación • Reducción dimensional: PCA, t-SNE, UMAP — visualización de alta dimensión • Detección de anomalías: Isolation Forest, LOF, Autoencoder	• Feature Engineering: encoding, binning, interacciones y transformaciones • Selección de features: Filter, Wrapper, Embedded (Lasso, importancia) • Manejo de datos desbalanceados, raros y de alta cardinalidad

Módulo 6.1 — Aprendizaje No Supervisado (10h)

- K-Means: inercia, Elbow Method, Silhouette Score y limitaciones
- DBSCAN: parámetros eps y min_samples — ideal para clusters de forma arbitraria
- Clustering jerárquico: dendrograma, linkage y corte óptimo
- PCA: varianza explicada, componentes principales y biplot de interpretación
- t-SNE y UMAP: visualización de embeddings en 2D — diferencias y cuándo usar cada uno
- Detección de anomalías: Isolation Forest en datos de transacciones financieras

Módulo 6.2 — Feature Engineering Avanzado (10h)

- Encoding categórico: One-Hot, Label, Target Encoding, WOE para alta cardinalidad
- Transformaciones numéricas: log, Box-Cox, Yeo-Johnson, binning y winsorizing
- Creación de features temporales: lag features, rolling statistics, tendencia y estacionalidad
- Interacciones y features polinomiales: PolynomialFeatures y feature crossing
- Selección automática: RFE, SelectFromModel y feature importance de permutación
- Feature Store conceptual: reutilización de features entre modelos en producción

PRÁCTICAS DEL NIVEL

P1 Segmentación de clientes con K-Means y DBSCAN — visualización PCA/UMAP y perfil de segmentos con acción de negocio

P2 Detección de fraude con Isolation Forest: dataset de tarjetas de crédito, threshold óptimo e informe de alertas

P3 Feature engineering competitivo: mejorar baseline de Nivel 5 en al menos 5% con nuevas features — comparativa

NIVEL 7 20 Horas | 3 Prácticas

Deep Learning y Redes Neuronales

Objetivo: Comprender y aplicar redes neuronales profundas con TensorFlow/Keras y PyTorch para visión artificial, series temporales y datos tabulares complejos.

■ ML / IA	■ Programación / Herramientas
• Arquitecturas: Dense, CNN, RNN, LSTM, Transformers (introducción) • Regularización DL: Dropout, Batch Normalization, Early Stopping, L2 • Transfer Learning: fine-tuning de modelos preentrenados (ResNet, EfficientNet)	• TensorFlow/Keras: Sequential API, Functional API y callbacks • PyTorch: tensores, autograd, DataLoader y training loop manual • GPU con Google Colab: CUDA, mixed precision y gestión de memoria

Módulo 7.1 — Redes Densas y CNN (10h)

- Neurona artificial: función de activación, pesos y bias — backpropagation intuitivo
- MLP (Multilayer Perceptron): arquitectura, activaciones (ReLU, Sigmoid, Softmax)
- Keras Sequential: construir, compilar, entrenar y evaluar en 10 líneas
- CNN: convolución, max pooling, flatten y arquitecturas para clasificación de imágenes
- Transfer Learning con VGG16/ResNet50: fine-tuning para dataset personalizado
- Data Augmentation: ImageDataGenerator y Albumentations para aumentar el dataset

Módulo 7.2 — RNN, LSTM y Series Temporales (10h)

- RNN: dependencias temporales, vanishing gradient y por qué falla en secuencias largas
- LSTM y GRU: puertas, memoria a largo plazo y aplicaciones en texto y series temporales
- Pronóstico de series temporales: ARIMA vs. LSTM — cuándo usar cada enfoque
- PyTorch training loop: Dataset, DataLoader, optimizer.zero_grad(), loss.backward()
- Weights & Biases (W&B;): tracking de experimentos, visualización de métricas y comparativas
- Hugging Face introductorio: cargar modelos preentrenados para clasificación de texto

PRÁCTICAS DEL NIVEL

P1 Clasificador de imágenes con CNN: construir desde cero en Keras y mejorar con Transfer Learning (ResNet50)

P2 Predicción de ventas con LSTM: serie temporal mensual, sliding window, evaluación y comparativa vs. ARIMA

P3 Proyecto integrador DL: elegir entre visión artificial, NLP básico o series temporales — pipeline completo con W&B;

NIVEL 8 20 Horas | 3 Prácticas

NLP: Procesamiento del Lenguaje Natural

Objetivo: Aplicar técnicas modernas de NLP con Hugging Face, transformers y LLMs para clasificación de texto, análisis de sentimiento, extracción de información y RAG.

■ ML / IA	■ Programación / Herramientas
• Transformers: BERT, RoBERTa, GPT — arquitectura y casos de uso • Fine-tuning: adaptar modelos preentrenados a tareas específicas • RAG (Retrieval-Augmented Generation): búsqueda semántica + LLM	• Hugging Face: pipeline(), Trainer API y datasets Hub • spaCy: tokenización, NER, POS tagging y dependency parsing • LangChain básico: chains, agentes y memoria para aplicaciones con LLMs

Módulo 8.1 — NLP Clásico y Transformers (10h)

- Preprocesamiento de texto: tokenización, stopwords, stemming, lemmatización
- Representaciones: Bag of Words, TF-IDF, Word2Vec, GloVe — evolución conceptual
- Clasificación de texto: análisis de sentimiento, detección de spam y categorización
- NER (Named Entity Recognition): extraer personas, organizaciones, fechas y cifras
- BERT y fine-tuning: tokenizer, dataset de Hugging Face y Trainer API
- Evaluación de NLP: F1, exacta match, BLEU, ROUGE — cuándo usar cada métrica

Módulo 8.2 — LLMs Aplicados y RAG (10h)

- Arquitectura Transformer en profundidad: atención multi-cabeza, embeddings posicionales
- Prompt Engineering para Data Science: extraer datos, clasificar y resumir documentos
- Embeddings vectoriales: sentence-transformers, similitud coseno y búsqueda semántica
- RAG pipeline: chunking → embeddings → vector DB (FAISS/Chroma) → retrieval → LLM
- LangChain: chains, retrieval QA y agentes para análisis de documentos corporativos
- Evaluación de LLMs: hallucinations, fidelidad, relevancia y benchmarks 2026

PRÁCTICAS DEL NIVEL

P1 Sistema de análisis de sentimiento fine-tuneado con BERT sobre reseñas del sector — F1 >90%

P2 Pipeline RAG completo: indexar 50 documentos PDF, búsqueda semántica y Q&A; conversacional con LangChain

P3 Extractor de información de contratos: NER personalizado + LLM — extraer partes, montos, fechas y alertas

NIVEL 9 20 Horas | 3 Prácticas

MLOps: Despliegue y Ciclo de Vida de Modelos

Objetivo: Llevar modelos de ML desde notebooks a producción real usando MLflow, FastAPI, Docker y GitHub Actions, con monitoreo continuo de rendimiento y drift.

■ Programación / Herramientas	■ ML / IA
• MLflow: tracking, registro, versionado y serving de modelos • FastAPI: crear API REST para servir predicciones en tiempo real • Docker: containerizar el pipeline completo de ML	• Monitoreo de modelos: data drift, concept drift y alertas • CI/CD para ML: GitHub Actions, tests automáticos y deploy continuo • Feature Store: Feast — centralizar y reutilizar features en producción

Módulo 9.1 — Empaquetado y Serving de Modelos (10h)

- MLflow: mlflow.log_params(), log_metrics(), log_model() y Model Registry
- Comparativa de runs: seleccionar el mejor modelo y promoverlo a producción
- FastAPI: endpoint /predict, validación con Pydantic, documentación automática (Swagger)
- Docker: Dockerfile para ML, imagen con dependencias y pruebas de integración
- Docker Compose: orquestar API + base de datos de predicciones en local
- Cloud deployment básico: AWS SageMaker endpoint o Google Cloud Run

Módulo 9.2 — CI/CD y Monitoreo de Modelos (10h)

- GitHub Actions: pipeline automático de test → train → validate → deploy al push
- Tests para ML: test de datos (Great Expectations), test de modelo y smoke tests
- Data drift: Evidently AI — detectar cuando los datos de producción cambian vs. entrenamiento
- Concept drift: cuando el target cambia — estrategias de reentrenamiento automático
- Logging y observabilidad: Prometheus, Grafana y alertas de degradación del modelo
- A/B testing de modelos en producción: shadow mode y canary deployments

PRÁCTICAS DEL NIVEL

P1 MLflow tracking completo: entrenar 5 variantes del modelo, comparar en UI y registrar el ganador

P2 API de predicción con FastAPI + Docker: endpoint /predict, tests automáticos y deploy en Cloud Run

P3 Pipeline CI/CD completo en GitHub Actions: push → tests → retrain → validate → deploy automático

NIVEL 10 20 Horas | 3 Prácticas

Big Data: Procesamiento a Gran Escala

Objetivo: Procesar volúmenes masivos de datos con Apache Spark, orquestar pipelines con Airflow y construir arquitecturas de datos modernas (Data Lake, Data Warehouse, Lakehouse).

■ Programación / Herramientas	■ Matemáticas / Estadística
• Apache Spark: RDDs, DataFrames, Spark SQL y MLlib • Airflow: DAGs, operadores, sensores y backfill de pipelines • dbt avanzado: modelos, macros, tests y documentación en producción	• Arquitecturas de datos: Lambda, Kappa, Medallion (Bronze/Silver/Gold) • Data Warehouse vs. Data Lake vs. Lakehouse — trade-offs 2026 • Streaming con Kafka + Spark Structured Streaming — ventanas y watermarks

Módulo 10.1 — Apache Spark y Procesamiento Distribuido (10h)

- Spark arquitectura: Driver, Executors, Partitions y el plan de ejecución
- PySpark: SparkSession, DataFrames, transformaciones lazy y acciones
- Spark SQL: consultas sobre billones de registros, funciones de ventana y UDFs
- Spark MLlib: pipeline de ML distribuido — clasificación y clustering a escala
- Delta Lake: ACID transactions, time travel y schema evolution sobre Spark
- Databricks Community Edition: cluster gratuito para prácticas a escala

Módulo 10.2 — Orquestación y Arquitecturas Modernas (10h)

- Apache Airflow: DAG, TaskGroup, XCom, BranchPythonOperator y sensores S3
- Pipeline de datos completo: ingestión → transformación dbt → validación → carga → alerta
- Data Lakehouse con Delta Lake: arquitectura medallion y consultas con Spark
- Kafka introductorio: productores, consumidores, topics y streaming de eventos
- Spark Structured Streaming: leer de Kafka, ventanas deslizantes y escritura en Delta
- Cost engineering: optimizar Spark jobs, particionamiento y gestión de costes en cloud

PRÁCTICAS DEL NIVEL

P1 Análisis de 100M+ registros de logs con PySpark en Databricks: agregaciones, ventanas y exportar resultado

P2 DAG de Airflow con 6 tareas: ingestión API → dbt transform → validación → carga BigQuery → notificación

P3 Pipeline de streaming: productor Kafka → Spark Structured Streaming → Delta Lake → dashboard en tiempo real

NIVEL 11 20 Horas | 3 Prácticas

IA Generativa y LLMs Aplicados a Data Science

Objetivo: Integrar LLMs y IA generativa en el flujo de trabajo del Data Scientist: generación de código, análisis conversacional de datos, agentes autónomos y fine-tuning.

■ ML / IA	■ Programación / Herramientas
• Fine-tuning de LLMs: LoRA, QLoRA y PEFT para datasets pequeños • Agentes de datos: LLM + herramientas (SQL, Python, búsqueda) para análisis autónomo • Multimodal: GPT-4o, Gemini Vision — analizar imágenes, tablas y gráficos	• Claude/OpenAI API: llamadas, streaming, function calling y manejo de costes • LangGraph: agentes con estado, ciclos y orquestación multi-agente • Código generado por IA: GitHub Copilot, Claude para Data Science

Módulo 11.1 — LLMs para el Flujo de Trabajo del Data Scientist (10h)

- Prompt Engineering avanzado para DS: generar código Pandas, SQL y visualizaciones
- Claude/GPT-4o API: llamadas programáticas, streaming, tool use y gestión de tokens
- Text-to-SQL con LLMs: convertir preguntas en lenguaje natural a consultas SQL precisas
- Análisis conversacional de datos: agente que responde preguntas sobre un DataFrame
- Code Interpreter / Advanced Data Analysis: el LLM ejecuta código Python en sandbox
- Automatización de reportes: LLM redacta narrativa ejecutiva desde métricas y gráficos

Módulo 11.2 — Fine-tuning y Agentes Autónomos (10h)

- Fine-tuning con LoRA/QLoRA: adaptar LLM a dominio específico con GPU pequeña
- Evaluación de LLMs: RAGAS, LLM-as-judge y benchmarks para tareas de DS
- LangGraph: agente de análisis con ciclo Observar → Planificar → Actuar → Reflexionar
- Agente de Data Science: genera código → ejecuta → interpreta resultado → itera
- MultiModal con Gemini Vision: extraer datos de imágenes, gráficos y tablas escaneadas
- Guardrails y seguridad: detectar alucinaciones, validar outputs y evitar prompt injection

PRÁCTICAS DEL NIVEL

P1 Sistema Text-to-SQL: interfaz en Streamlit donde el usuario pregunta en lenguaje natural y obtiene resultados de la BD

P2 Agente de análisis autónomo con LangGraph: recibe dataset → propone hipótesis → genera código → reporta hallazgos

P3 Fine-tuning LoRA de un LLM para clasificación de documentos del sector — comparativa antes/después en benchmark propio

NIVEL 12 20 Horas | 3 Prácticas

Proyecto Final: Data Science en Producción

Objetivo: Integrar todas las competencias del diplomado en un proyecto de Data Science completo, de extremo a extremo, desplegado en producción y presentado ante un panel evaluador.

■ ML / IA	■ Programación / Herramientas
• Ciclo completo: problema de negocio → datos → EDA → modelo → deploy → monitoreo • Documentación técnica: tarjeta del modelo, limitaciones y sesgos identificados • Presentación ejecutiva: ROI del modelo, métricas de negocio y plan de mantenimiento	• Repositorio profesional: estructura, README, requirements.txt y Docker • Demo en producción: API activa, dashboard y reporte automático funcionando • Portfolio público: GitHub, Kaggle y caso de uso en LinkedIn

Módulo 12.1 — Diseño y Desarrollo del Proyecto Final (12h)

- Definición del problema: enmarcarlo como problema de DS con impacto de negocio medible
- Recopilación y validación de datos: fuentes, calidad, linaje y documentación
- EDA y Feature Engineering: aplicar todo lo aprendido con justificación de cada decisión
- Modelado y experimentación: comparar mínimo 3 enfoques en MLflow, seleccionar y optimizar
- Despliegue: API con FastAPI, containerización con Docker y CI/CD con GitHub Actions
- Monitoreo: dashboard de métricas, alerta de drift y plan de reentrenamiento automático

Módulo 12.2 — Presentación y Evaluación (8h)

- Estructura del informe final: resumen ejecutivo, metodología, resultados y conclusiones
- Tarjeta del modelo (Model Card): propósito, datos, métricas, limitaciones y sesgos
- Demo en vivo: presentar el sistema funcionando en producción ante el panel evaluador
- Defensa técnica: responder preguntas sobre decisiones metodológicas y alternativas
- Feedback cruzado: revisión de proyectos de compañeros con metodología estructurada
- Portfolio y siguientes pasos: publicar en GitHub, Kaggle, HuggingFace Hub y LinkedIn

PRÁCTICAS DEL NIVEL

P1 Prototipo funcional del proyecto: EDA completo + modelo baseline + API inicial desplegada

P2 Sistema en producción: modelo optimizado + API robusta + CI/CD + monitoreo activo + dashboard

P3 PRESENTACIÓN FINAL: demo en vivo + informe técnico + model card + repositorio público profesional

Perfil de Egreso y Roles Profesionales

Al completar el Diplomado, el participante estará capacitado para desempeñarse en los principales roles del ecosistema de datos:

Rol Profesional	Niveles Requeridos	Salario referencial 2026
Data Analyst	Niveles 1–4	USD 35k–60k/año
Data Scientist Junior	Niveles 1–6	USD 55k–85k/año
Data Scientist Senior	Niveles 1–8	USD 85k–130k/año
ML Engineer	Niveles 1–9	USD 90k–140k/año
Big Data Engineer	Niveles 1–4, 10	USD 80k–125k/año
NLP / LLM Engineer	Niveles 1–8, 11	USD 95k–150k/año
MLOps Engineer	Niveles 1–9	USD 90k–145k/año
Data Science Lead	Niveles 1–12 (completo)	USD 120k–180k/año

Certificaciones Internacionales Relacionadas

Certificación	Organismo	Niveles del Diplomado
IBM Data Science Professional Certificate	IBM / Coursera	Niveles 1–5
Google Professional Data Engineer	Google Cloud	Niveles 3–10
AWS Certified Machine Learning Specialty	Amazon Web Services	Niveles 5–9
Databricks Certified Associate Developer (Spark)	Databricks	Nivel 10
TensorFlow Developer Certificate	Google / TensorFlow	Niveles 7–8
Hugging Face NLP Course Certificate	Hugging Face	Nivel 8
Microsoft DP-100: Azure Data Scientist	Microsoft	Niveles 5–9
Kaggle Competitions Expert / Master	Kaggle	Niveles 1–8
MLflow Certified Practitioner	Databricks	Nivel 9
Professional Data Scientist (PDS)	DASCA	Niveles 1–12 (completo)

Stack Tecnológico del Diplomado

Categoría	Herramientas	Niveles
Lenguajes	Python 3.12, R, SQL, Bash	1–12
Entornos	Jupyter, VS Code, Google Colab, Databricks Community	1–12
Datos y EDA	Pandas, NumPy, SciPy, Polars, Ydata-Profiling	1–4
Visualización	Matplotlib, Seaborn, Plotly, Tableau, Power BI, Looker Studio	4, 12
ML Clásico	Scikit-learn, XGBoost, LightGBM, CatBoost, Optuna	5–6
Deep Learning	TensorFlow, Keras, PyTorch, Hugging Face, W&B;	7–8

NLP / LLMs	spaCy, NLTK, Transformers, LangChain, LangGraph, FAISS	8, 11
IA Generativa	Claude API, OpenAI API, Gemini API, Ollama, LoRA/QLoRA	11
MLOps	MLflow, FastAPI, Docker, GitHub Actions, Evidently AI, Prefect	9
Big Data	Apache Spark, PySpark, Airflow, Kafka, Delta Lake, dbt	10
Bases de datos	PostgreSQL, MySQL, BigQuery, MongoDB, Redis, Pinecone	3, 10
Cloud	AWS (S3, SageMaker, Redshift), GCP (BigQuery, Vertex AI), Azure ML	9–12

Al completar el Diplomado en Data Science (Edición 2026), el participante habrá desarrollado las competencias completas para extraer conocimiento de valor, construir modelos predictivos e implementar soluciones de IA en producción en cualquier organización o proyecto propio.

www.uneweb.edu.ve