

Alexis Alejandro Martínez Suárez

alexis.martinez.6584@gmail.com | +56 9 6847 9046 | [linkedin/alexismartinez](https://www.linkedin.com/in/alexismartinez/) | [github/alex-msu](https://github.com/alex-msu)

PERFIL

Estudiante de **Ingeniería en Informática** con enfoque en **Ciencia de Datos**, **Machine Learning** y desarrollo de soluciones basadas en **Python**.

EDUCACIÓN

Ingeniería en Informática, mención en Data Science
DUOC UC – PLAZA OESTE

Cerrillos, RM | Mar 2022 – Actualidad

Técnico en Administración de Empresas (mención RRHH)
COLEGIO COMERCIAL DE PEÑAFLO

Peñaflor, RM | Titulado 2020

CERTIFICACIONES

TOEIC | 980/990 – NIVEL C1 DE INGLÉS (MCER)

ETS | Dic 2024

PROYECTOS

PREDICCIÓN DE RETENCIÓN DE CLIENTES (CHURN) CON MLOPS

PYTHON, SCIKIT-LEARN, MLFLOW,
PANDAS, MATPLOTLIB, ARGPARSE

Desarrollé un pipeline de machine learning modular y reproducible para predecir el churn bancario, estructurado con buenas prácticas de MLOps.

Implementé limpieza de datos, ingeniería de atributos, escalado, codificación categórica, entrenamiento y evaluación del modelo (Random Forest).

Integración completa con MLflow para registrar métricas, artefactos, parámetros y versiones del modelo.

Alta prioridad en recall y F1-score para minimizar falsos negativos.

Resultado: F1-score y recall ≥ 0.96 .

PREDICCIÓN DE LLUVIA CON ML + APP WEB

PYTHON, PANDAS, SCIKIT-LEARN,
FLASK, HTML, PICKLE

Desarrollé un modelo de clasificación para predecir lluvia utilizando datos meteorológicos de Australia. Implementé una aplicación web usando Flask para probar el modelo.

Resultado: Precisión de 88% en datos de prueba.

ANÁLISIS DE SENTIMIENTO CON RNN, LSTM Y TRANSFORMER

PYTHON, KERAS, TENSORFLOW,
SCIKIT-LEARN, PYTORCH, NLP

Clasifiqué sentimientos en tweets usando el dataset Sentiment140. Comparé desempeño entre modelos RNN y LSTM mediante precisión, F1-score y pérdida, seleccionando LSTM como arquitectura final. También desarrollé una implementación educativa de un Transformer básico en PyTorch para explorar su arquitectura.

Resultado: LSTM superó a RNN en desempeño (77% frente a 72% de precisión)— el Transformer se integró como experimento conceptual.

CLASIFICACIÓN DE IMÁGENES CON TRANSFER LEARNING

PYTHON, TENSORFLOW, KERAS,
MATPLOTLIB

Diseñé y entrené una CNN con aprendizaje por transferencia para clasificar imágenes de CIFAR-10. Implementé técnicas de regularización (Dropout, L2, Data Augmentation) que redujeron el overfitting y mejoraron la generalización.

Resultado: 89.04% de precisión en validación.

HABILIDADES TÉCNICAS

Lenguajes

Python, SQL, Java, JavaScript, HTML, CSS.

Ciencia de Datos y ML

Pandas, NumPy, scikit-learn, Matplotlib, Seaborn. Ingeniería de atributos, selección de variables y uso de métricas de clasificación como precision, recall y F1-score.

MLOps

Entrenamiento y evaluación reproducible de modelos. Uso de **MLflow** para el seguimiento de métricas, parámetros y artefactos. Organización modular de scripts en Python, uso de **argparse** para pipelines CLI, trabajo con entornos virtuales (**venv**) y versionado con Git/GitHub.

Deep Learning

TensorFlow, Keras, PyTorch.

Herramientas y Entornos

Git, GitHub, Jupyter Notebook, Google Colab (entorno en la nube), conocimientos básicos en Docker, Power BI, Notion.

Bases de Datos

MySQL, SQLite.

Otros

Microsoft Excel, Word, PowerPoint.

HABILIDADES BLANDAS

Comunicación

Comunicación clara y efectiva en entornos colaborativos.

Resolución de Problemas

Pensamiento analítico orientado a la resolución de problemas complejos.

Aprendizaje

Enfoque en mejora continua y autoaprendizaje constante.

Motivación

Alta motivación por aprender y aportar en proyectos interdisciplinarios.

Adaptabilidad

Capacidad de adaptación a metodologías ágiles y nuevas tecnologías.

TEMAS DE INTERÉS ACTUAL

Prompt Engineering

Diseño y optimización de instrucciones para controlar y mejorar el rendimiento de modelos de lenguaje.

Retrieval-Augmented Generation (RAG)

Integración de recuperación de información con generación de texto para mayor precisión contextual.

Automatización de pipelines (CI/CD)

Implementación de flujos automáticos de entrenamiento, evaluación y despliegue de modelos.

Monitorización de modelos

Seguimiento del comportamiento de modelos en producción, detección de drift y alertas de desempeño.