

■ TEMARIO COMPLETO

Data Science y Dashboards Interactivos

Guía de Estudio Desde Cero Hasta Nivel Profesional

Autor: Kevin Vargas

LinkedIn: www.linkedin.com/in/kevargas

Fecha: September 2025

■ Descripción del Temario:

Este temario te guiará paso a paso para aprender a crear dashboards profesionales como el proyecto GDP World Economic Analysis. Está diseñado para llevarte desde los fundamentos básicos hasta un nivel avanzado, con recursos prácticos y proyectos reales.

■ **Objetivo:** Al completar este temario, serás capaz de crear dashboards interactivos profesionales utilizando Python, Pandas, Plotly y Streamlit de forma completamente independiente.

■■ **Tiempo estimado:** 6-12 meses (dedicando 1-2 horas diarias)

■ FASE 1: FUNDAMENTOS DE PROGRAMACIÓN

1.1 Python Básico (4-6 semanas)

- Sintaxis básica y variables
- Tipos de datos (int, float, string, boolean)
- Estructuras de control (if/elif/else, loops)
- Funciones y parámetros
- Listas, tuplas y diccionarios
- Manejo de errores (try/except)
- Comprensión de listas
- Módulos e importaciones

■ Recursos recomendados:

- Libro: 'Automate the Boring Stuff with Python' (gratuito online)
- Curso: Python.org Tutorial oficial
- Práctica: HackerRank Python Domain
- Plataforma: Codecademy Python Course

1.2 Python Intermedio (3-4 semanas)

- Programación orientada a objetos (clases, métodos)
- Decoradores y funciones lambda
- Manejo de archivos (CSV, JSON, TXT)
- Bibliotecas estándar importantes (os, datetime, json)
- Expresiones regulares (regex)
- Generadores e iteradores
- Context managers (with statements)

■ Recursos recomendados:

- Libro: 'Effective Python' by Brett Slatkin
- Curso: Real Python membership
- Documentación: docs.python.org

■ FASE 2: FUNDAMENTOS DE DATA SCIENCE

2.1 NumPy - Computación Numérica (2-3 semanas)

- Arrays de NumPy vs listas de Python
- Creación de arrays (zeros, ones, arange, linspace)
- Indexing y slicing avanzado
- Operaciones matemáticas y broadcasting
- Funciones estadísticas (mean, std, max, min)
- Álgebra lineal básica
- Reshape y manipulación de dimensiones
- Máscaras booleanas y filtrado

■ Recursos recomendados:

- Documentación oficial: numpy.org
- Curso: DataCamp NumPy
- Tutorial: NumPy Quickstart tutorial

2.2 Pandas - Manipulación de Datos (4-5 semanas)

- DataFrames y Series: conceptos fundamentales
- Lectura de datos (CSV, Excel, JSON, APIs)
- Indexing y selección (.loc, .iloc, [])
- Limpieza de datos (valores faltantes, duplicados)
- Transformación de datos (apply, map, lambda)
- Groupby y agregaciones avanzadas
- Merging, joining y concatenación
- Pivoting y reshaping (pivot, melt)
- Manejo de fechas y series temporales
- Operaciones con strings
- Optimización de memoria y performance

■ Recursos recomendados:

- Libro: 'Python for Data Analysis' by Wes McKinney
- Documentación: pandas.pydata.org
- Curso: Kaggle Learn Pandas
- Práctica: 10 Minutes to Pandas (oficial)

2.3 Matplotlib - Visualización Básica (2-3 semanas)

- Conceptos básicos: Figure, Axes, Artist
- Tipos de gráficos básicos (line, scatter, bar, histogram)
- Customización (colores, estilos, etiquetas)
- Subplots y múltiples gráficos
- Anotaciones y texto
- Guardado de figuras en diferentes formatos
- Estilos predefinidos y personalización avanzada

■ Recursos recomendados:

- Documentación: matplotlib.org
- Tutorial: Matplotlib tutorials oficiales
- Libro: 'Matplotlib for Python Developers'

■ FASE 3: VISUALIZACIÓN AVANZADA

3.1 Plotly - Gráficos Interactivos (3-4 semanas)

- Plotly Express vs Plotly Graph Objects
- Gráficos básicos interactivos (scatter, line, bar)
- Customización avanzada (hover, colores, temas)
- Gráficos complejos (subplots, multiple axes)
- Mapas y geovisualizaciones
- Animaciones y transiciones
- 3D plotting y surface plots
- Dash integration básico
- Export y sharing de gráficos

■ Recursos recomendados:

- Documentación oficial: plotly.com/python/
- Tutorial: Plotly Fundamentals
- Ejemplos: Plotly Gallery

3.2 Seaborn - Visualización Estadística (2 semanas)

- Statistical plots (distplot, boxplot, violin)
- Correlation matrices y heatmaps
- Categorical plots
- Regression plots
- Styling y themes
- Integration con matplotlib

■ FASE 4: APLICACIONES WEB

4.1 Streamlit - Dashboards Interactivos (4-5 semanas)

- Conceptos básicos: `st.write`, `st.title`, `st.header`
- Widgets interactivos (selectbox, slider, multiselect)
- Layout y containers (columns, sidebar, expander)
- Caching y performance (`@st.cache_data`)
- File uploads y downloads
- Forms y user input
- State management (`session_state`)
- Custom components y HTML
- Deployment (Streamlit Cloud, Heroku)
- Configuration y theming

■ Recursos recomendados:

- Documentación: docs.streamlit.io
- Tutorial oficial: 30 Days of Streamlit
- Gallery: streamlit.io/gallery
- YouTube: Streamlit channel

4.2 HTML/CSS Básico - Personalización (2-3 semanas)

- HTML básico (tags, structure, forms)
- CSS básico (selectors, properties, box model)
- CSS Grid y Flexbox
- Responsive design básico
- Bootstrap framework
- Integration con Streamlit (`unsafe_allow_html`)

■ Recursos recomendados:

- Curso: freeCodeCamp HTML/CSS
- Referencia: MDN Web Docs
- Práctica: CSS Grid Garden, Flexbox Froggy

■ FASE 5: MACHINE LEARNING APLICADO

5.1 Scikit-learn - ML Básico (4-5 semanas)

- Conceptos fundamentales (supervised vs unsupervised)
- Preprocessing (scaling, encoding, feature selection)
- Regression (linear, polynomial, ridge, lasso)
- Classification (logistic, SVM, random forest)
- Clustering (K-means, hierarchical)
- Model evaluation (train/test split, cross-validation)
- Metrics (accuracy, precision, recall, R^2)
- Pipeline creation y automatización
- Feature engineering básico

■ Recursos recomendados:

- Documentación: scikit-learn.org
- Libro: 'Hands-On Machine Learning' by Aurélien Géron
- Curso: Coursera Machine Learning (Andrew Ng)

5.2 Time Series Analysis (2-3 semanas)

- Time series basics (trends, seasonality, noise)
- Prophet library para forecasting
- ARIMA models básico
- Visualization de series temporales
- Evaluation metrics para forecasting
- Deployment de modelos de predicción

■ Recursos recomendados:

- Documentación: facebook.github.io/prophet/
- Tutorial: Time Series Analysis with Python

■ FASE 6: MANEJO DE DATOS Y APIs

6.1 APIs y Web Scraping (2-3 semanas)

- REST APIs básico (GET, POST requests)
- Libraries: requests, urllib
- Authentication (API keys, OAuth básico)
- JSON parsing y handling
- Web scraping con BeautifulSoup
- Rate limiting y best practices
- Error handling para APIs
- Caching de data from APIs

6.2 Bases de Datos (2-3 semanas)

- SQL básico (SELECT, WHERE, JOIN, GROUP BY)
- SQLite integration con Python
- PostgreSQL básico
- SQLAlchemy ORM básico
- Database design básico
- Data warehousing concepts

■ Recursos recomendados:

- Curso: SQLBolt (interactive SQL tutorial)
- Documentación: requests.readthedocs.io
- Tutorial: Beautiful Soup documentation

■ FASE 7: DEPLOYMENT Y PRODUCTION

7.1 Git y GitHub (1-2 semanas)

- Git basics (init, add, commit, push, pull)
- Branching y merging
- GitHub workflow
- README.md y documentation
- Version control best practices
- GitHub Pages for portfolio

7.2 Cloud Deployment (2-3 semanas)

- Streamlit Cloud deployment

- Heroku deployment básico
- Environment variables y secrets
- Requirements.txt y dependency management
- Docker básico (opcional pero recomendado)
- Domain setup y custom URLs

■ PROYECTOS PRÁCTICOS SUGERIDOS

Proyecto 1: Dashboard de Finanzas Personales

Descripción: Crear dashboard para tracking de gastos con upload de CSV

Skills aplicados: Pandas, Plotly, Streamlit, File handling

Proyecto 2: Análisis de Datos de Ventas

Descripción: Dashboard para analizar ventas con KPIs y forecasting

Skills aplicados: Data analysis, Time series, Prophet, Visualización

Proyecto 3: Dashboard de Crypto/Stocks

Descripción: Real-time dashboard con APIs financieras

Skills aplicados: APIs, Real-time data, Caching, Advanced Plotly

Proyecto 4: ML Model Dashboard

Descripción: Interface para entrenar y evaluar modelos ML

Skills aplicados: Scikit-learn, Model evaluation, Interactive ML

Proyecto 5: Portfolio completo

Descripción: Website personal con todos los proyectos

Skills aplicados: GitHub Pages, Documentation, Professional presentation

■ RECURSOS ADICIONALES RECOMENDADOS

Libros Esenciales:

- Python for Data Analysis (Wes McKinney)
- Hands-On Machine Learning (Aurélien Géron)
- Storytelling with Data (Cole Nussbaumer Knaflic)
- The Visual Display of Quantitative Information (Edward Tufte)

Plataformas de Práctica:

- Kaggle - Datasets y competitions
- DataCamp - Cursos interactivos
- Coursera - Certificaciones
- GitHub - Portfolio y proyectos
- Stack Overflow - Resolución de dudas
- Real Python - Tutoriales avanzados

Comunidades y Networking:

- LinkedIn Data Science groups
- Reddit: r/datascience, r/MachineLearning
- Discord: Python, Data Science servers
- Local meetups y conferences
- Twitter: Sigue a data scientists influyentes

■ CRONOGRAMA SUGERIDO

Meses 1-2: Python Básico e Intermedio + NumPy

Meses 3-4: Pandas + Matplotlib + Proyecto 1

Meses 5-6: Plotly + Seaborn + Streamlit Básico

Meses 7-8: Streamlit Avanzado + HTML/CSS + Proyecto 2

Meses 9-10: Machine Learning + Time Series + Proyecto 3

Meses 11-12: APIs + Databases + Deployment + Proyectos 4-5

■ TIPS PARA EL ÉXITO

- **Practica diariamente:** Dedicar al menos 1-2 horas diarias, la consistencia es clave
- **Build projects:** No solo estudies teoría, crea proyectos reales desde el inicio
- **Documenta tu progreso:** Usa GitHub para mostrar tu evolución
- **Join communities:** Conecta con otros learners y profesionales
- **Don't get overwhelmed:** Es normal sentirse abrumado, toma breaks cuando sea necesario
- **Focus on fundamentals:** Domina bien lo básico antes de avanzar
- **Learn by teaching:** Explica conceptos a otros o escribe blogs
- **Stay updated:** La tecnología cambia rápido, mantente al día

¡Éxito en tu journey de Data Science! ■

Kevin Vargas - LinkedIn: www.linkedin.com/in/kevargas