DATA SCIENCE FESTIVAL

MAYDAY

18.05.24

MLOPS in Financial Services

Michelle Conway

Lead Data Scientist,
@ Lloyds Banking Group

LLOYDS BANKING GROUP

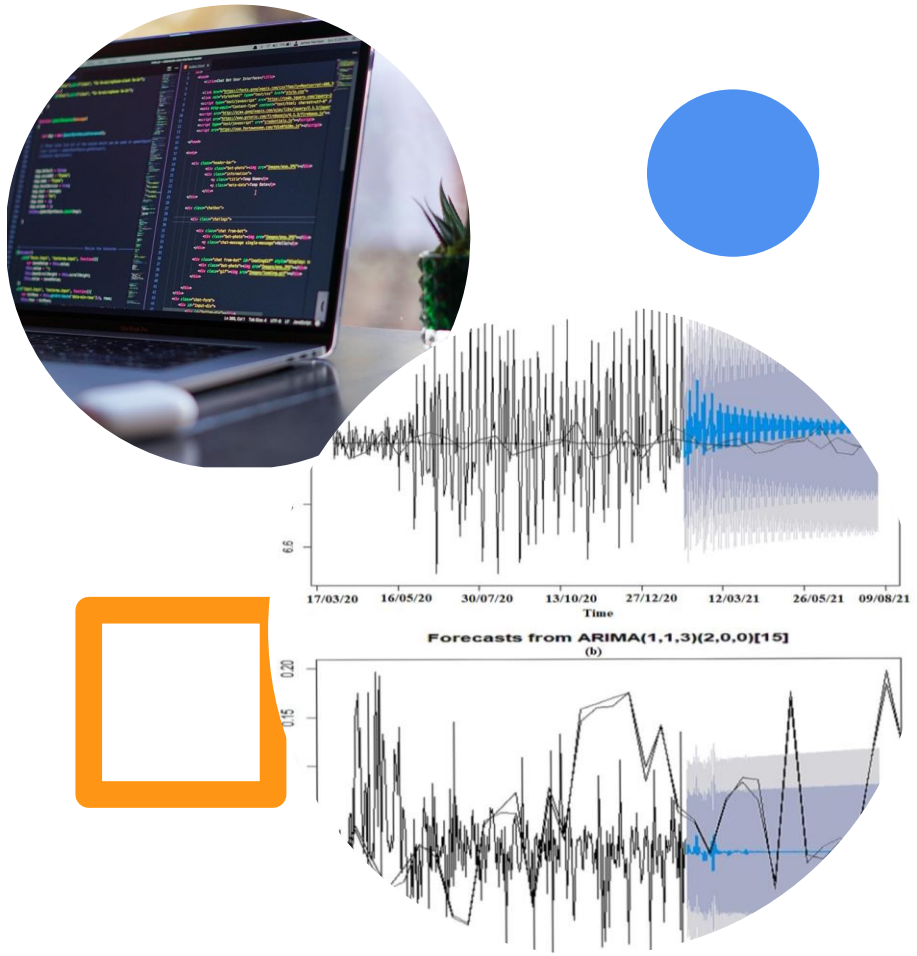
Agenda

- Financial Forecasting
- ARIMA Modelling
- Github Demo on Python Libraries
- Python Libraries for Production

Introduction

Financial forecasting plays a crucial role in maintaining **regulatory compliance** and **managing risks** effectively in the Financial Services sector.

ML Ops, or **Machine Learning Operations**, streamlines the deployment and management of data science models in **production environments.**



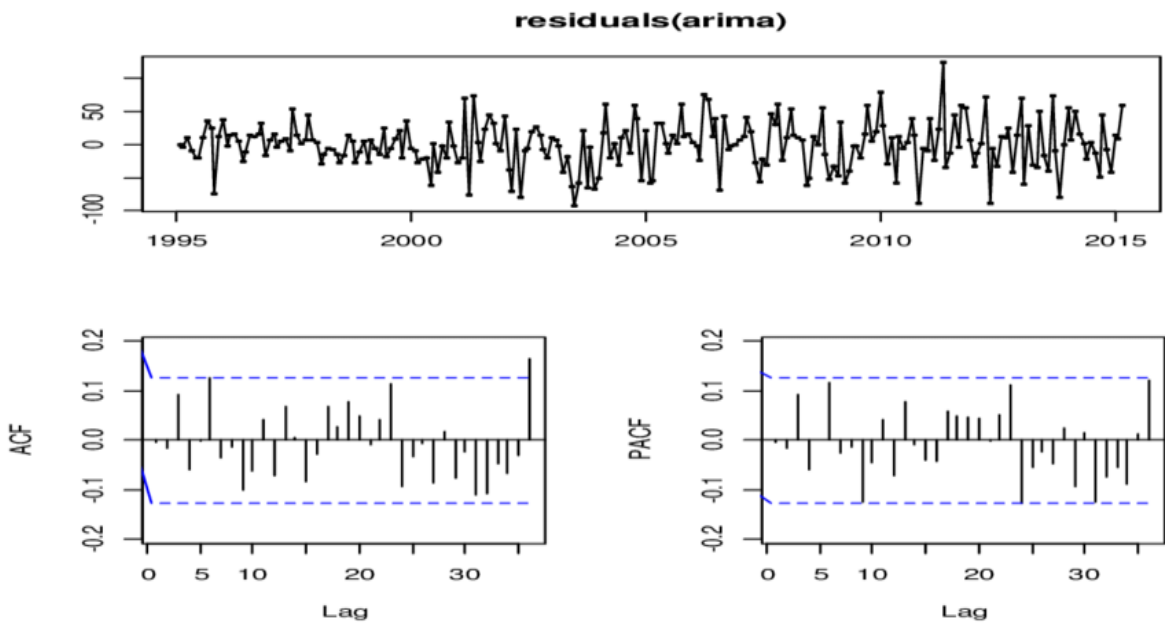
ARIMA Modeling (AutoRegressive Integrated Moving Average)

AutoRegressive:

Represents the correlation between a variable and its past values or "lags" .

Integrated (I)

Differencing of raw observations to make the time series stationary.



ARIMA(p, d, q) model consists of **three components**:

- **p**: Number of lag observations included in the model (AR order).
- **d**: Degree of differencing (I order).
- **q**: Size of the moving average window (MA order).

Moving Average (MA)

Captures the dependency between an observation and a residual error from a moving average model.

ACF & PACF

The selection of appropriate lag orders (p and q) is often determined through methods like **autocorrelation function (ACF)** and **partial autocorrelation function (PACF)** plots.

ARIMA Modelling Python Libraries

https://github.com/mconwa02/dsf-may-day-2024

- **Statsmodels:** Provides comprehensive tools for statistical modeling and forecasting.
- **Sktime**: (Scikit time series) specialized library for time series forecasting, offering a unified interface for various forecasting algorithms.
- **Darts:** (Data Analytics and Regression Testing Suite) Library for probabilistic and deterministic time series forecasting, suitable for complex financial data.

Data Science Production Projects with Python

Fire - CLI

- **Fire** simplifies command-line interface (CLI) creation for easy model deployment and execution.

https://pypi.org/project/fire/

```
import fire

def hello(name="World"):
  return "Hello %s!" % name

if __name__ == '__main__':
  fire.Fire(hello)

python hello.py  # Hello World!
python hello.py --name=DSF  # Hello DSF!
```

Loguru - Logging

- Utilising **Loguru** for comprehensive logging and monitoring in production environments.

https://pypi.org/project/loguru/

```
from loguru import logger

logger.debug("That's it," /
             "beautiful and" /
             "simple logging!")
```

Codebase Design

- Importance of **modular code** and **Unit testing**, especially adopting test-driven design (TDD).
- **Ruff** extremely fast Python linter and code formatter, written in Rust.
- 10-100x faster than existing linters (like Flake8) and formatters (like Black)
- https://pypi.org/project/ruff/

Thank you

Connect with me on LinkedIn

Michelle Conway

Lead Data Scientist 🏆Twenty in Data & Tech 2023 by Women in Dat...