## Python/R Software Engineer | Contract Data Scientist | Statistician | AI/Machine Learning Specialist

1040, Brussels, Belgium
rohail.taimour@gmail.com | +32 489 83 64 76 | Personal website | Linkedin | Github

# Rohail Taimour

## Summary

As a seasoned Python/R Software Engineer and Data Scientist with a Master's degree in Statistics, I specialize in creating robust data products with a focus on machine learning. My dynamic skill set bridges the gap between data science, machine learning engineering, and data engineering, allowing me to thrive in fast-paced, ambiguous environments. I am passionate about enhancing project efficiency and impact through best practices and agile methodologies.

## Education

**MSc in Statistics** - KU Leuven, Leuven, Belgium - 2014-2016

Graduated Cum Laude, Master's thesis on continuous optimization of production processes in MATLAB

**BSc (Hons.) in Accounting and Finance** - Lahore University of Management Sciences (LUMS), Lahore, Pakistan - 2010-2014

Graduated with Distinction (3.6/4.0)
Courses: Operations Research, Supply Chain, Decision analysis, Applied Probability
Treasurer for University Adventure Society organizing hiking trips for groups of upto 300 people in North of Pakistan

## Technical competencies

| | |
|---|---|
| **Programming Languages:** | Expert in Python and R with 7+ years of experience |
| **Operating Systems:** | Proficient with Windows, MacOS, Linux (Redhat/Ubuntu) |
| **Development Environments:** | Pycharm, VScode, Rstudio, Jupyter Notebooks, Azure Databricks |
| **CI/CD:** | Git, GitHub Actions, Azure Pipelines, GitLab Pipelines, GitHub CLI |
| **Containerization:** | DockerHub, Docker |
| **Cloud Services:** | AWS (S3, ECS), Azure (Blob, Databricks, Pipelines) |
| **Database Management:** | PostgreSQL, SQLite3, Neo4j, SQLAlchemy |
| **Technical Documentation:** | Pandoc, Markdown, |

## Freelance projects (Oct 2022-present)

### Automated SQL Script Generation for Cross-Platform Data Migration in PostgreSQL

**Python Software Engineer, Illumina, Mechelen, Belgium**  July-Aug 2023

- Designed and implemented an ORM Mapper for dynamic ingestion of various file formats, automating the SQL script generation process for data migration.
- Implemented the solution as a Python package encapsulating the entire data migration logic within a Docker entrypoint for portability and ease of deployment.
- Conducted comprehensive testing of generated SQL scripts using mock PostgreSQL database tables, ensuring script accuracy and reliability.
- Parameterized key inputs allowing for seamless deployment across multiple environments (Development, Integration, Production).

### Multi-Layered Python Solution for Bioinformatics Pipeline Management and Automation

**Python Software Engineer and Data Pipeline Architect, Illumina, Mechelen, Belgium**  April 2023 - October 2023

- Implemented a Python service to routinely monitor new sequencing data, tracking progress using an SQLite database.
- Developed a multi-layered Python package: one layer encapsulated a data processing component, packaged within a **Docker** runtime environment, for handling bioinformatics pipeline outputs.
- Created a separate entry point within the package responsible for initiating and managing the Docker-packaged data processing component, as well as other routine operations such as downloading completed analyses, updating analyses statuses, etc
- Implemented comprehensive systems integration, utilizing a combination of CLI tools and API calls for effective coordination and automation across various software components.
- Applied Object-Oriented Programming (OOP) techniques to organize API, database interactions and endpoint processing.
- Implemented unit testing using **pytest** and implemented fail-safe mechanisms for robust error handling.

### Design and implement information retrieval methods using Natural language processing (NLP)

**Machine Learning Engineer, IT Supply Quality, GSK Belgium**  Oct 2022-Feb 2023

- Improved performance of information retrieval by 20% on unseen test data using a custom named entity recognition (NER) from **Spacy**.
- Performed POC's on Azure DataBricks environment to improve model performance using rule-based techniques as well as **NER** and annotated data to train custom NER.
- Added text preprocessing features to the NLP pipeline such as **Spacy** tokenization, Part of speech (POS) tagging, better handling of non-english emails, breaking emails into sentences, etc.

| | Sphinx for documentation; proficient in CSS, HTML for web development |
|---|---|
| **Data Science:** | Pandas, PyMC3, scikit-Learn, sktime, Seaborn |
| **Data Engineering:** | Kedro, Prefect, PySpark |
| **Software Testing:** | Pytest for testing; Black, Pre-Commit, iSort, Flake8, Mypy for code quality |
| **Package Management:** | Conda, Mamba, Pip, Poetry for Python package and environment management |
| **Machine Learning Operations (MLOPs):** | MLflow for artifact management and model versioning |

## Personal details

- Nationality: Belgian, Pakistani
- Languages: English (fluent/bilingual), Urdu (Native), French (B1)
- Mobility: Driving License available, flexible for hybrid setup with up to 4 days on site
- Availability: Immediately
- Hobbies: Drumming and percussion instruments, Bouldering/Climbing, productivity, Squash, reading

# Data science projects at IT AI team, UCB Pharmaceutical (2016-Oct 2022)

## Yield optimization for batch and continuous production processes using Machine Learning in Python

**Lead Data Scientist, Supply and Manufacturing, UCB Switzerland/Belgium**     Aug 2020-Oct 2022

- Production setting proposed by model directly led to an increased throughput of 20%, turning in a recurring 1.5 million euro in annual cost savings
- Analyze time series data collected from equipment sensors and visually summarize golden batch insights
- Created (Bayesian) and tree-based regression models to quantify impact of process changes and predict batch performance
- Performed a thorough model validation and hyperparameter tuning exercise before recommending model insights be tested in a live production environment
- Supported delivery of workshops demystifying the process of conducting AI projects and machine learning to process experts

## Python Framework for Customized Promotional Responsiveness Models Across Regions

**AI/ML engineer, Lead Data Scientist, Go to Market/Commercial EU5, US and Japan, UCB**     June 2019-June 2021

- Developed a Python package with **Cookiecutter** templates that abstract the complexities of the data science workflow, enabling configurable deployments across diverse scenarios such as different countries and disease areas.
- Enhanced the package to seamlessly wrap over **scikit-learn**, thereby simplifying key data science tasks from preprocessing to model training and tuning
- Incorporated **MLflow** into the package for robust artifact management, allowing for the tracking of model versions, data inputs, and predictions
- Created customer segmentation models and proposed optimal resource allocation based on customer responsiveness to different marketing channels
- Investigated adaptations to data science methodology for country/product specificities for maximum reusability. Delivered as many as ten different use cases for different products and countries
- Performed feature engineering using **PySpark** and validated ingested data using data visualization methods and discussions with subject-matter experts

## Scientific influencer (KOLs) identification, ranking and profiling using network analytics and Neo4j

**Data scientist/Product owner, Drug Development, Commercial, Medical affairs, UCB**     2018-2019

- Developed custom **Neo4j** databases integrating diverse data sources for KOL influence analysis, enhancing data-driven decision-making.
- Utilized **py2neo** within **Jupyter Notebooks** for interactive data manipulation and network visualizations, employing tools like NetworkX and **Cytoscape** for insightful analysis.
- Interacted with the Graph database via **Cypher** queries in the web UI as well as via the CLI for data extraction, exploration and reporting.
- Supported improvements in the intake of customer requests to reduce time to deliver reports from days to hours

## Developed an Automated Forecasting Workflow of Claims Data from US Healthcare System

**Lead Data Scientist, US Finance and claims, UCB**     2017-2018

- Engineered specialized **R packages** focusing on separate concerns: data engineering for preprocessing, a wrapper over Facebook's Prophet for advanced forecasting, and automated

reporting for performance analysis.

- Designed and implemented a comprehensive end-to-end workflow for ingesting healthcare claims data, performing time-series forecasting, and generating insightful reports on forecasting accuracy.
- Achieved over 90% forecasting accuracy across various use cases by meticulously tuning models and integrating bespoke anomaly detection algorithms for time series data.
- Conducted extensive hyperparameter tuning and model validation using **high-performance computing** to optimize forecasting models effectively.
- Automated report generation using **R Markdown**, providing clear, concise insights into forecasting performance and model accuracy.

### Hands-on workshop to demystify Artificial intelligence and Machine Learning

**Data science instructor, IT departments US, EU, UCB**                    May- June 2017

- Created a R shiny application to create an engaging way for participants to learn about typical AI use cases
- Delivered the workshop to over 100 people in four different venues and received great feedback on level of engagement

## Personal Projects

### Web Scraper to analyse Property Purchase and Rental Trends in Belgium

- Developed web scraper using Beautiful Soup to collect information such as apartment data such as price, area, etc.
- Implemented SQLite for data storage, using `**Pydantic**` for data validation and `**SQLAlchemy**` for database interactions.
- Encapsulated the concerns into a python package with dependency management using Poetry.
- Employed **Prefect** for job orchestration, managing the workflow's scheduling and monitoring of scraping tasks.

### Personal Portfolio and blogging website built using Hugo and hosted using Github Pages

- Created website using `Hugo` and implemented features such as a contact form, and visitor commenting capabilities.
- Hosted the static website on GitHub Pages and automated the deployment process using GitHub Actions.
- Codebase hosted on github

### Automated Resume Builder and Continuous Deployment System with GitHub Pages Hosting

- Engineered an automated system for generating, versioning, and hosting a dynamic CV using Markdown, HTML, Jinja templating and CSS.
- Set up a trio of GitHub repositories to separately manage the CV's content, styling, and public hosting on Github Pages.
- Developed a Python package for automating the styling and generation of the CV, integrating with Markdown and HTML/CSS.
- Implemented version control for CV content using a private GitHub repository, ensuring secure and organized data management.
- Leveraged GitHub Actions for automating the CV's generation and deployment process, enabling updates through git pushes.
- Hosted the final CV on GitHub Pages, providing a live, online version that can be easily updated

### Unit Commitment Solver for Power Grid Optimization via FastAPI

- Developed a REST API using **FastAPI** for optimizing energy distribution among powerplants based on load requirements and fuel costs.
- Implemented multiple algorithms to solve the **unit-commitment problem**, considering factors like fuel cost, powerplant efficiency, and environmental constraints.
- Utilized **Pydantic** for data validation and schema definition, ensuring data integrity and streamlined request handling.

- Deployed the API service using **Uvicorn**, an ASGI server, to ensure high performance and concurrent request handling.
- Packaged and containerized the application using **Docker**, with detailed documentation and a Dockerfile for easy deployment and scalability.
- Employed **pytest**, along with Python best practices such as typing and linting..
- Managed project dependencies using **Poetry**, facilitating efficient workflow and package management.
- Integrated **Swagger UI** for interactive API documentation and testing, accessible via http://localhost:8888/docs.



- Deployed the API service using **Uvicorn**, an ASGI server, to ensure high performance and concurrent request handling.
- Packaged and containerized the application using **Docker**, with detailed documentation and a Dockerfile for easy deployment and scalability.
- Employed **pytest**, along with Python best practices such as typing and linting..
- Managed project dependencies using **Poetry**, facilitating efficient workflow and package management.
- Integrated **Swagger UI** for interactive API documentation and testing, accessible via http://localhost:8888/docs.