Adrian Ng MSc.
Graduate Data Engineer

Email: adrian.j.ng@gmail.com
Mobile: +447766336972
Website: https://adrian.ng

SUMMARY

Computer Science graduate seeking junior-level Data Engineering roles. In particular: opportunities that may apply and grow my *Java* skillset, which I used to build my dissertation project and have been working with ever since. Prior to postgraduate study, my technical expertise was in *SQL development*. My recent role as a *Data Analyst* proved to be a brief but beneficial learning experience in which I found that problems of a technical nature are far more stimulating than analytical ones. Thus naturally follows a career in programming.

EDUCATION

- **Master of Science in Data Science and Analytics** — with Distinction
  Department of Computer Science, Royal Holloway, University of London — *Sept. 2016 – Dec. 2017*

- **Bachelor of Engineering in Mechanical Engineering** — Upper Second Class with Honours
  School of Engineering, King's College London, University of London — *Sept. 2007 – July 2010*

JAVA PROJECTS

- **Dissertation: Implementation of Value at Risk (VaR) measures in Java**
  (https://adrian.ng/java/var/)   (https://github.com/Adrian-Ng/VaR)
  This project implements various approaches to estimating *VaR*, a measure of risk. These are: *Model Building*, *Historical Simulation*, and *Monte Carlo Simulation*. In addition, the following approaches to estimating market volatility are implemented: *Equal Weighted*, *Exponentially Weighted Moving Average*, and *GARCH(1,1)*.
    - **Object Oriented Design**
      As we have a number of approaches to estimating both *VaR* and variance, abstract methods *getVariance()* and *getVaR()* are defined in abstract classes but implemented in child classes. Instantiation of these child classes is handeled via the *factory method*.
    - **Concurrency**
      The *Monte Carlo* approach generates a large number of random walks, which takes a long time in series. These are parallelized using *java.util.concurrent* for which *Callable* and *Future* lambdas are implemented.
    - **Data Ingress**
      Real-world market data is sourced using the *Yahoo Finance API*.
      A hypothetical portfolio with a number of stocks and options is defined locally across a numberof text-files.

- **OptionPricer**
  (https://adrian.ng/java/options/)   (https://github.com/Adrian-Ng/OptionPricer)
  The implementation of three approaches to estimating option prices in Java: *Monte Carlo simulation*, *Black-Scholes equations*, and*Binomial Trees*. Estimates are returned via *getCall()*, *getPut()* interface methods. Abstract classes were utilised when implementing different ways of estimating the same thing (e.g. European vs American vs Asian payoff). *CallableFutures*

- **Data Mining with Apache Hadoop**
  Twitter Data, Enron Corpus, regex, *map reduce*. Hadoop programs written in Java. E.g. extracting the communications network from *Enron Corpus*, aggregation of Twitter data in *JSON* format. Programs were tested on local installation of a single-node *HDFS*.

- **Google PageRank**
  This project involved the implementation in Java of Google's *PageRank* algorithm which simulates the behaviour of someone browsing a series of webpages by computing a transition matrix from an input graph and mixing a Markov Chain.

POSTGRADUATE COURSEWORK: IMPLEMENTATION OF MACHINE LEARNING ALGORITHMS

- **R Language**:

k-Nearest Neighbours	LDA/QDA	Kernel Methods
Neural Network regression	Decision Trees	Hierarchical Clustering

- **MATLAB**:

Hidden Markov Models	Aggregating Algorithm

Professional Experience

- **Manchester City Football Club** London
  Data Analyst *Jan. - July 2018*

  ○ **NYCFC Project**
  This SQL data engineering project involved the implementation of processes to automate the ingress of external data sources (transactional data from Ticketmaster, customer data from NYCFC). Conferencing with stakeholders (New York City FC) and partners (Major League Soccer) was necessary to understand the *data situation*.

  ○ **GDPR Customer Preferences**
  This project involved the creation of a number of automated processes to *MERGE* data regarding GDPR preferences with our database.

  ○ **Tableau Dashboard Automation/Optimisation**
  Implemented *Data Cubes* to pre-aggregate data along all possible subset of categorical fields. Front-end dashboards retained their exploratory flexibility but removed real-time computational burden from front-end which improved UX.

  ○ **Guiding and Mentoring**
  Instructing junior colleagues on SQL Server Management Studio best practices and fundamentals. E.g. understanding DDL & DML for writing SQL queries and creating database objects; when to return a *product join* vs *semi-join*; making use of *information_schema*; utilising *SQL Agent* to schedule jobs. Scheduled and ad-hoc workshop sessions were scheduled to provide this instruction.

- **ITG Creator** Westminster
  Senior CRM Campaign Executive *Dec. 2013 - Sept. 2016*

  ○ **Virgin Media Processes**
  Built an automated segmentation process using SQL stored procedures for other members of the team to utilise. Recipient data were imported via *BULK INSERT*, stored in DB tables and indexed. The output being segmentation data to be linked to HTML content and broadcast to recipients.

  ○ **Advanced Queries**
  Used recursive queries (CTEs) to clean data e.g. removing n-number of leading zeros from mobile phone numbers in order to prefix with dialling codes. Or splitting delimited strings into table format.

  ○ **Query Execution**
  Improved cross-server query execution speeds by using *OPENQUERY*, which transmits a SQL query in string format to the remote server. Dynamic SQL was utilised to include XML data in the string. Using a *CTE*, this XML could be represented as relational object. Effectively this allowed us to join and filter on the remote server using data from the local sever. As a result, only a small data set was returned via the *OPENQUERY*. Details on website.

  ○ **Soft Skills**
  Attended inter-departmental work assessment groups and advised on job specifications. As senior team member, served as point of contact for clients and colleagues looking to resource our team. Jira was used to monitor workload and queue work to team members.

- **Languages**: Java 8, T-SQL

- **Software**: IntelliJ IDEA, SQL Server Management Studio, Sublime, Git, Jira, Maven

- **GitHub**: https://github.com/Adrian-Ng