**Adrian Ng MSc.** Email: contact@adrian.ng
**Seeking Junior-Level Data Engineering Opportunities** Mobile: +44-77-66-33-6972

## Summary

I am a Computer Science graduate passionate about Data Engineering. I thrive when writing code, building things, and solving technical problems. I seek opportunities that further my growing experience in *Java* – a language which I used in numerous academic projects ranging from the implementation of financial models to large-scale data processing with applications written for *Apache Hadoop MapReduce*.

Prior to postgraduate study, my expertise was in *SQL development*, primarily focused on the implementation of data segmentation processes for digital marketing communications. My recent role as a Data Analyst at *Manchester City FC* was a brief but beneficial learning experience: confirmation that my abilities lay more in the technical than the analytical. Thus naturally I persue 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

- **Implementation of Value at Risk (VaR) measures in Java** (https://adrian.ng/java/var/) (https://github.com/Adrian-Ng/VaR)
  This dissertation 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 variance were 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()` were each defined in abstract an class but implemented in numerous child classes. Instantiation of these was handled via the *factory method*.
  - **Concurrency**
    The *Monte Carlo* approach generates a large number of random walks, which can take a long time to fully execute in series. Therefore execution is parallelized using `java.util.concurrent` for which `Callable` and `Future` lambdas were implemented.
  - **Data Ingress**
    Real-world market data was sourced using the *Yahoo Finance API*, which returns a collection of `BigDecimal`. These daily closing prices are transformed into daily price changes of type `Double[]` – thus sacrificing accuracy for computational performance.
    Other inputs, such as the make-up of our hypothetical investment portfolio were defined locally in text files to be read via `Scanner in`.
- **Option Pricing** (https://adrian.ng/java/options/) (https://github.com/Adrian-Ng/OptionPricer)
  This project implements three approaches to estimating option prices in Java: *Monte Carlo simulation*, *Black-Scholes equations*, and*Binomial Trees*.
  Option prices for each of these approaches were retrieved through the interface methods `getCall()` and `getPut()`. Abstract classes were also used to facilitate the implementation of various real-world option types, which each differ in how *pay-off* is calculated.
- **Data Mining with Hadoop MapReduce** (https://github.com/Adrian-Ng/HadoopEnron)
  A number of *MapReduce* applications were written in Java with a variety of purposes: e.g. extracting the communications network from the *Enron Corpus* or aggregation of Twitter data in *JSON* format.
  These applications extended `Mapper` and `Reduce` classes. Methods `map()` and `reduce()` were overridden with bespoke implementation.
  Java artifacts were exported and executed on Hadoop clusters (both single node and distributed). Input/Output data were stored in HDFS and accessed via `hadoop fs` commands.
  A subsequent exercise was undertaken to minimise the verbosity of these *Hadoop MapReduce* applications by translating them to *Scala* for use in a *Spark REPL*.
- **Java 8 Streams with financial data** (https://adrian.ng/java/yahoofinance/#stream)
  A small exercise involving the use of *Java 8 Streams*. Processing real-world financial data to return *mean* and *equal-weighted variance* of some market asset by defining functions to `map()` and `reduce()` methods.

## Professional Experience

- **Manchester City Football Club** Euston, London
  *Data Analyst – Fan Relationship Management* *Jan. - July 2018*
  - **NYCFC Project**
    This data engineering project involved the implementation of SQL *stored procedures* to automate the ingress of data from external sources (transactional data from Ticketmaster, customer data from NYCFC).
    Conference calls with stakeholders (New York City FC) and partners (Major League Soccer) were regularly hosted/attended to understand the *data situation*.
  - **GDPR Customer Preferences**
    This project involved the creation of a number of automated processes to merge GDPR preference data with the analytical database. To accomplish this, SQL DML such as `merge` was utilised.
  - **Tableau Dashboard Automation/Optimisation**
    Implemented *Data Cubes* to pre-aggregate data along all combinations of categorical fields. That is, every possible drill-down and roll-up was computed in advance. As a result, front-end dashboards retained their exploratory flexibility but removed real-time computational burden. Thus improving user-experience.
  - **Guiding and Mentoring**
    Instructing junior colleagues on SQL 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.
    Both regular and ad-hoc workshop sessions were held in order to provide this instruction.
- **ITG Creator (Digital Marketing Agency)** Westminster, London
  *Senior CRM Campaign Executive – SQL Development* *Dec. 2013 - Sept. 2016*
  - **Segmentation Processes**
    Built a number of automated segmentation process using SQL stored procedures for team members to utilise. Recipient data were imported via `BULK INSERT`, stored in database tables and indexed (clustered). Segmentation data was output and linked to HTML content to be broadcast to recipients.
  - **Debugging**
    Try/Catch blocks Logging Error printing (if erroring while attempted to execute dynamic sql, print SQL)
  - **Recursion** (https://adrian.ng/SQL/cte/Recursion/)
    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 strings and mapping into relational format.
  - **Cross-Server Query Optimisation** (https://adrian.ng/SQL/misc/openquery-xml)
    Improved cross-server query execution speeds by using `OPENQUERY`, which transmits a string of SQL for execution on the remote (a live database under constant heavy load). Futher, *Dynamic SQL* was utilised to include XML data in the string. Mapping via a *CTE*, this XML could be transformed into a relational object capable of joining to remote objects. As a result, filtering via join occurs remotely and only a small data set is returned via the `OPENQUERY`.
  - **Soft Skills**
    - Attended inter-departmental work assessment groups and advised on work specifications.
    - As senior team member, served as point of contact for clients and colleagues looking to resource our team.
    - On occasion I held responsibility for resourcing and managing the team's workload using *Jira*.
- **Seatwave (now Ticketmaster)** Moorgate, London
  *Marketing Analyst Intern – Commercial Team* *May 2013 - Dec. 2013*
  - **Basic SQL**
    In this position I gained my first experience writing database queries in *SQL Server Management Studio*. With basic understanding of *DML* and *DDL*, I was able to query the ticketing and customer databases to extract data for warehousing, analysis, and CRM segmentation.

## General

| | |
|---|---|
| **Languages:** | Java 8, T-SQL |
| **Software:** | IntelliJ IDEA, SQL Server Management Studio, Sublime, Git, Jira, Maven |
| **Look at my code:** | https://github.com/Adrian-Ng |
| **Discussion of my code:** | https://adrian.ng |