Important information for UCT's **MPhil specialising in Financial Technology**

Convener: A/Prof Co-Pierre Georg (co-pierre.georg@uct.ac.za)

First things first: there is a list of frequently asked questions at the end of this document. Before you send me an email, please check the list.

Important note for 2020: The degree is being moved within the Commerce Faculty for 2020 because I have been awarded the South African Reserve Bank Chair in Financial Stability Studies. Consequently, I have moved departments and will be based at the School of Economics going forward. The degree remains an MPhil in Financial Technology but we might change some of the courses and some courses will get new course codes.

Applications open 1 August and close 30 September. If you have applied before this date, please double-check that you have selected the correct registration (if you have registered for CM033BUS29 as degree code and DOC5032F, DOC5039F, DOC5037F, and DOC5005W as course codes, they are likely to change. Please check http://cogeorg.github.io in early August for an update).

Information technology is becoming pivotal for the financial services industry, which faces a changing skills need in order to deal with an increasingly complex regulatory framework and increased competitive pressure from fintech start-ups.

UCT through the Departments of Statistics and Computer Science, the School of Economics, and the UCT Financial Innovation Lab is ideally positioned to address the shortage of suitably trained graduates in this area. Based on an existing MSc specialising in Data Science, an interdisciplinary degree convened by the Department of Statistics, UCT has created this new degree in Financial Technology by offering three elective courses as well as a 60-credit dissertation convened by the Financial Innovation Lab. The course is designed as a part-time degree. It is suitable for continuing students, i.e. students who are currently working in the financial services industry who want to use the degree to learn the skills needed to adapt to a rapidly changing financial industry.

In this Master's degree, students acquire the necessary skills in both finance and computer science to start a career in the financial services industry. Specifically, the degree will cover the following courses:

Semester 1

DOC5032F South African Financial Markets AIFMRM (15 credits)
(will be replaced in 2020)

CSC5007Z Databases for Data Scientists	COMPSCI	(12 credits)
STA5075Z Stat. and High Performance Computing	STATS	(12 credits)
DOC5039F Financial Software Engineering	SOE	(15 credits)
(will have new course code in 2020)		
CSC5008Z Data Visualization	COMPSCI	(12 credits)
STA5076Z Supervised Learning	STATS	(18 credits)
STA5077Z Unsupervised Learning	STATS	(12 credits)

Semester 2

DOC5037F Fintech and Cryptocurrencies	SOE	(24 credits)
(will have new course code in 2020)		
DOC5005W Minor Dissertation	SOE	(60 credits)
(will have new course code in 2020)		

TOTAL: 180 credits

The Fintech and Cryptocurrencies course will be taught en bloc at the end of the first semester and all practicals for this course will be project focused.

Students are encouraged to collaborate with fintech startups in the vibrant Cape Town ecosystem and work on practical projects for their thesis. Through this collaboration we hope to instill an *entrepreneurial spirit* in the students. Because of the practical focus of the degree as a whole and the Minor Dissertation in particular, we envisage that some of the dissertation projects might lead to commercially viable ideas.

Cost:

Semester 1

DOC5032F South African Financial Markets	R 5,690
(will be replaced in 2020)	
CSC5007Z Databases for Data Scientists	R 6,590
STA5075Z Statistical and HPC	R 6,590
DOC5039F Financial Software Engineering	R 5,678
(will have new course code in 2020)	
CSC5008Z Data Visualization	R 6,590
STA5076Z Supervised Learning	R 9,720
STA5077Z Unsupervised Learning	R 6,590

Semester 2

DOC5037F Fintech and Cryptocurrencies (will have new course code in 2020)	R 9,084
DOC5005W Minor Dissertation (will have new course code in 2020)	R 9,040
TOTAL	**R65,572**

Information on the official UCT online application will be updated in this document by 15 August 2019. Please note that this degree is a 120 credit coursework and 60 credit thesis degree and that students will be awarded a Master of Philosophy upon completion.

Syllabi of existing courses

Databases for Data Scientists (CSC5007Z)

This course will introduce students with little or no prior experience to the three cornerstone database technologies for big data, namely relational, NoSQL and Hadoop ecosystems, The course aims to give students an understanding of how data is organised and manipulated at large scale, and practical experience of the design and development of such databases using open source infrastructure. The relational part will cover conceptual, logical and physical database design, including ER modelling and normalisation theory, as well as SQL coding and best practices for performance enhancement. NoSQL database were developed for big data and semi-structured data applications where relational systems are too inefficient; all four types of NoSQL architecture will be introduced. Distributed data processing is key in manipulating large data sets effectively. The final section of the course will teach the popular Hadoop technologies for distributed data processing, such as MapReduce programming and the execution model of Apache Spark.

Statistical and High Performance Computing (STA5075Z)

This course aims to provide students with a foundation in statistical computing for data science. The course is divided into three sections, namely Basic Programming, High Performance Computing and Simulation & Optimisation. In the first section students will learn how to write computer programs to analyse data with the R Language and Environment for Statistical Computing. Students will then be taught how to run jobs in parallel on a remote computer cluster using a Linux command prompt. Finally the course will introduce students to the fundamental principles and uses of simulation and optimisation.

Data Visualization (CSC5008Z)

Visualization is the graphical representation of data with the goal of improving comprehension, communication, hypothesis generation and decision making. This course aims to teach the principles of effective vascularization of large, multidimensional data sets. We cover the field of visual thinking, outlining current understanding of human perception and demonstration how we can use this knowledge to create more effective data visualizations.

Supervised Learning (STA5067Z)

Supervised learning is a set of statistical modelling tools for predicting or estimating the relationships between predictor and target variables in complex data sets. As part of the Masters in Data Science degree this course aims to familiarise students with the statistical methodology needed to analyse the relationships between predictor and target variables in big data. The students should be able to apply the appropriate statistical methods such as Generalized Linear Models, Tree-Based Methods, Multivariate Methods, Feature Extraction, Support Vector Machines and Neural Networks to analyse a big data set and estimate the relationships between the predictor and target variables.

Unsupervised Learning (STA5077Z)

This course aims to familiarise students with the statistical methodology needed to analyse relationships between variables in big data without having causal relationships with predictor and response variables. Topics covered include association rules and market basket analysis, self-organising maps, multidimensional scaling, cluster analysis, principal component analysis.

Syllabi of new courses

DOC5039F Financial Software Engineering

(will have new course code in 2020)

This course will provide a thorough introduction to the python programming language as well as the most important libraries that are used in the financial services sector. These include numpy and scipy, as well as an in-depth look at the pandas module which is the most commonly used module for data analytics. The course will start with an introduction to python data structures before introducing functions and then classes.

The course will then take some time to explain recursions and introduce a number of elementary algorithms and data management techniques including merge sort, heap sort, binary search trees, breadth- and depth-first search and finding shortest paths using the Dijkstra algorithm.

Developing object-oriented projects in python will be the main focus of this course. Concepts such as inheritance and abstract base classes will be introduced, and will complement the practical introduction to python with a theoretical overview of these concepts. Students will also learn how to write re-usable code, how to document code properly, and how to use unit-testing and test-driven development to ensure high code quality. These methods are crucial in organizations with a large existing code base and will ensure that students know current best practices of software development.

All tools are introduced theoretically, but the course places substantial emphasis on practical applications. At every point in the course students will put their theoretical knowledge into practical applications. The course will focus on financial applications wherever possible.

Key outcomes:
1. Exhibit mastery of basic principles of software engineering on both a theoretical and practical level.
2. Understand and be able to apply advanced software engineering methods used to develop complex software.
3. Being able to develop complex software in python.

DOC5037F Fintech and Cryptocurrencies
(will have new course code in 2020)
Emerging platforms and decentralised technologies provide new ways to transfer value, aggregate and analyse information, improving connectivity and reducing the marginal costs of accessing information and participating in financial activities. The course will enable students to harvest the power of these new platforms and decentralised technologies.

This course is comprised of three parts, each accounting for roughly one third of the course. The first part will introduce the regulatory environment banks and non-bank financial intermediaries operate in. This includes global regulatory minimum standards such as Basel III, as well as the domestic regulatory environment. This framework serves as a backdrop to understand the recent changes in the financial services industry and the challenge posed by disruptive fintech innovations.

The second part is concerned with modern financial market infrastructure and in particular distributed ledger technologies as basis for these disruptive innovations. The material will cover a basic introduction to different financial instruments and how they are traded in modern financial systems. These include equities traded on exchanges, bonds and securities traded in over-the-counter markets, as well as interbank loans settled on payment systems. Traditional settlement methods and double entry bookkeeping are then contrasted with distributed ledger technologies. The focus of this part of the course is to enable students to gain a thorough understanding of the blockchain and distributed ledger technologies, including an introduction to the necessary foundations in cryptography. This introduction will include cryptographic hash functions, hash pointers, Merkle trees, digital signatures, and public-key cryptography.

We will then turn to distributed consensus, and consensus without identity before discussing proof-of-work, proof-of-stake, and proof-of-authority methods as well as other distributed consensus mechanisms that ensure the integrity of the blockchain. The course will be wrapped up with a practical example of a cryptocurrency using the Ethereum blockchain. Students will be able to write simple applications on Ethereum using smart contracts.

Key outcomes:

1. Understand the regulatory environment in which financial institutions operate and how this shapes fintech applications.
2. Master the concepts of distributed ledgers both in theory as well as in practical applications.
3. Understand cryptocurrencies both in theory as well as in practical applications
4. Being able to develop simple cryptocurrency applications.

Additional Information

Entrance requirements

A relevant Honours degree or equivalent (equivalent to a UCT Hons degree), with at least a good 2nd class pass (above 65%). Note that the 65% threshold is not a binding restriction.

Finance

UCT has a number of scholarships available. Check UCT website for opportunities.

Frequently Asked Question

Q: Can I apply via email?
A: Note that we cannot accept applications via email. As indicated on our website, you must apply online through the application form:
http://www.students.uct.ac.za/students/applications/apply/forms
Every applicant must complete the official online application form above.

Q: I do not have a four year degree, can I still apply?
A: You need a four year degree (e.g. a Bachelors + Honours) to be eligible for this Masters, i.e. an NQF level 8 qualification. Unfortunately, we cannot count work experience towards your qualifications.

Q: I do not have a background in computer science or statistics, can I still apply?
A: The requirement of having a background in computer science and statistics is not a binding one. We are interested in top students from a variety of backgrounds. You will need a healthy dose of mathematics, though, and must have at least a strong desire to learn how to program.

Q: I do not have 65% in my Honours, can I still apply?
A: The requirement of having 65% is not a binding one. Note, however, that admission into the course will be highly competitive.

Q: Should I submit my CV as part of the online application?
A: Yes. We will still consider your application if the CV is missing, but to give us a full overview of your background, please include it. If you forgot to submit your CV and you cannot change your application, please send it to co-pierre.georg@uct.ac.za with your full name and student number (if available).

Q: What level of mathematics and coding do I need?
A: In terms of math, you will need some basic calculus (in particular you need to understand what a gradient is) and statistics. We don't per se require you to have taken coding classes at university before, but ideally you have coding experience or at the very least a strong interest in learning how to code.

Q: I want to do the degree part-time. When are classes and do I need to be based in Cape Town?
A: The program can be done both full time and part-time, but you will need to be in Cape Town either way since classes will be in the morning or late afternoon. There is also some teaching en bloc from mid June to mid July. For both full time and part-time students the degree will be one year since part-time students will have some professional experience already which should make it a bit easier for them to

cope with the workload. We do not have the full schedule yet but will update this document as soon as we have.

Q: My coding is a bit rusty, what can I do to brush it up?
A: One way of doing this is to do some of the online courses offered by codecademy, udemy, or coursera. We will be using a lot of python in the degree. Check https://github.com/cogeorg/teaching and in particular our wikis for extensive reading lists and a collection of self-study materials that will help you be optimally prepared.

Q: How many students will you admit to the program?
A: We will be admitting no more than 15 students in 2020. The program will be heavily oversubscribed. However, we will give every single application our full consideration and want to encourage all interested students to apply.