Detailed Work Experience

Tarak Kharrat

February 14, 2020

Abstract

In this document, I give a detailed overview of my work experience. It is meant to complement my short resume, giving the reader more information about the type of projects I have been involved in, and the tools I have built during my career.

Contents

1 Overview

I am a data scientist with extensive experience of building **predictive models** in competitive commercial environments. I have an undergraduate degree in pure mathematics, a masters degree in quantitative finance and a PhD in statistics. Over the past 10 years, I have been working with data on a daily basis and developed a set of tools that allows me to help any business interested in making predictions.

2 Academia

After finishing my PhD in October 2015, I have been working as a (part-time) research fellow contributing to research papers in the field of computational statistics and sports analytics.

Although I like developing new methodologies in my research, I am keen to demonstrate new methodologies in a practical setting. Therefore, my papers either solve a real life problem or develop some new methodology to solve an existing problem. For example, the new family of counting processes described in [Baker and Kharrat, 2018] and the associated software [Kharrat et al., 2019] were motivated by the restrictive assumptions of the Poisson model systematically applied in the football forecasting literature. Thus, in [Boshnakov et al., 2017] we argued that the application of the new methodology (the renewal counting processes) gave a better prediction than the benchmark (the Poisson model and its derivatives).

In my academic journey, I have had the opportunity to collaborate with world class researchers from the United Kingdom and Canada and published papers in top journals. My full list of publication can be found below:

- **Applied mathematics**: [Heil et al., 2012]
- **Computational statistics**: [Baker and Kharrat, 2018, Kharrat et al., 2019]
- **Sports analytics**: [Boshnakov et al., 2017, Kharrat et al., 2020] and [Guiliang et al., 2020]

3 Industry

3.1 Trading (finance)

I started on the trading floor in April 2011 as an intern within the proprietary trading department of *Societe Generale* Paris before moving to New York in

July 2011 as a full time employee.

The main focus of my job was to use quantitative tools and data to help traders find alphas and design/refine semi or fully automated trading strategies. I collaborated closely with the high frequency and basket trading desks and worked mainly on the US Equity and Future markets (S&P500, Russell 1000 and to some extent Nasdaq).

The standard workflow usually begins with ideas/observations coming from the traders. I worked together with them on refining these ideas, transforming them into hypotheses to be tested using data.

The choice of the training data is vital for the success of such tasks and hence I used the traders expertise to carefully define the data samples before starting any modelling. In particular, special attention was given to check that the data generation process was likely to be similar (between our training data points and the future trading points). In other words, we had to carefully pick up (enough) data points from the past where market conditions were likely to be similar to current (and near future) market conditions.

Although details on the strategies cannot be fully disclosed here, I was mainly working on two major subjects: stocks correlation within an index and stock likelihood to join or leave an index (index re-balancing). I also worked occasionally on some ideas related to special events (mainly M&A) where we tried to estimate the probability of a deal to move from *announced* to *agreed* or from *completed* to *terminated*.

3.2 Betting (football)

The betting market offers a solid benchmark for anyone trying to forecast the outcome of a football match. Although one of my papers [Boshnakov et al., 2017] discussed this problem, I really started to properly look at this subject when I started my collaboration with Kickdex in late 2015. The mission was complex but clearly defined: building (from scratch) an accurate predictive model able to generate profit when used in an automated (algorithmic) betting strategy on the Asian market.

It took us slightly less than 4 years to reach that goal but it was a very enriching experience where I learned a lot about how predictive models work, how to improve them, interpret their errors and communicate research findings to non-technical domain experts. I also had the opportunity to work alongside very talented quants and IT experts who taught me a lot on daily basis.

In the rest of this section, I describe in more detail the steps we followed to build our cutting-edge advanced model:

3.2.1 Identify the appropriate data sources

The main feed provider used by Kickdex is Opta. The choice of the provider was a business decision made at the investors' level and hence changing it would require building a strong case.

The first decision we subsequently had to make was how to store the data. `MongoDB`, a `non-SQL` type database, was selected. The dataset was very unstructured and after consulting with the IT team the choice of `mongoDB` was straightforward.

The second step was to define a set of tests to apply to raw feed when received from the provider to ensure that every (game) feed entering the database is trustworthy. These tests check various properties of the data such as validity (11 players on the pitch, every player on the team sheet has at least one event associated to him,...), consistency (same id used to identify the same player,...) and absence of outliers (no games with a very high number of specific events such as shots for example). We were able using this process to flag corrupted games and report them to the provider asking for corrections when possible. At the end of this step, we were confident that every data point (a document in the `mongoDB` terminology) could be used for modelling.

We also identified a number of websites containing complementary information. The websites were found based on expert advise, from academic papers or from specialised forums. Once selected, the feed was extracted by web-scraping, cleaned, validated (using the same logic described above) and then mapped to the Opta feed and stored in our database.

3.2.2 Defining the right metric to evaluate a set of probabilities

With a valid dataset in hand, the next step was to define a method to evaluate the *quality* of a set of predictions i.e. probabilities produced for unseen games. The method had to be adapted to our application (the property of our betting strategy), and should have attractive mathematical properties (for example being a proper scoring rule). After studying what has been suggested in the literature, we found out that the ideas discussed in [Johnstone et al., 2011] were the more relevant for our objective and adapted them to build a *Kickdex proper scoring rule*.

3.2.3 Feature engineering

Features are the fuel of predictive models and hence feeding good features into a model has a dramatic impact on its performance. Designing (good)

features is more of an art than a scientific process and relies heavily on domain knowledge and the available data. We collected ideas (inspiration) from academic papers, specialised forums and discussion with professional punters to create our feature set. We carefully checked that our features did not include future information (also known as *look-ahead bias*) and tried carefully to avoid any systematic bias. We also tried to diversify our feature set as much as possible to capture the complexity of a football game. It is worth noting that a feature set is usually dynamic: new features get introduced regularly whilst others are rejected as the result of new research and development.

3.2.4 Feature selection

It is often the case when working with features to end up having a large pool of candidates to test. This usually happens because you may have designed several small variations of the same idea or because you wanted to capture different aspect of the game. Although some models can select features at the fitting/training step, others can fail to converge when presented with highly correlated features (linear models for example). Even if the model used is robust to highly correlated features, it is usually a bad idea to pass a large feature set as it will increase dramatically the time needed for training. Therefore, using a filter to keep only the relevant information is a crucial task when building a predictive model.

At Kickdex, we conducted extensive research on the subject and tested different approaches based on clustering, feature mirroring, ... We ended up building a propriety feature selection process that proved to be useful and allowed us to reduce the size of our set from several thousand features initially to less than 50.

3.2.5 Feeding the right features to the right learners

It is important to know how the feature values are going to be digested by a specific learner (i.e model) and it is sometimes useful to transform a feature before passing it. For example, models that estimates parameters by numerically minimising a loss function may benefit from features being on the same scale (neural networks are one family of such models).

At Kickdex, we spent time trying different transformations and how they performed when associated with specific learners. We built good understanding on what combinations work best and leveraged that when building our final model.

3.2.6 Combining the successful learners

After trying several modelling approaches, we came to the conclusion that no single model is likely to outperform all the others for the different type of markets we wanted to predict. Therefore, we decided to combine different type of learners using different sets of features to build an ensemble. The hope was that the sub-models would commit different type of errors (ideally uncorrelated) and hence combining them will result in better predictions (than the best sub model in the ensemble).

We designed a framework to train/test an ensemble of arbitrarily selected base learners on generic classification tasks using ideas presented in [Caruana et al., 2004]. We leveraged cloud computing power to test the performance of a candidate model on real data. After some code optimisation and careful redesign, we managed to reduce the training time from a few weeks to less than 10 hours for the current version used in production.

3.2.7 Explaining the predictions

Although our ensemble can be seen as a type of *black box* model, we attached specific care to its interpretability and tried as much as possible to explain its prediction. Using tools such as the `DALEX` package [Biecek, 2018], we tried to understand how the predictions changed with changes in feature values and for given prediction scenarios (usually when the model is far from the benchmark i.e the betting market) which features were responsible for the estimated probability. This approach allowed us first to build confidence on our predictions and second to get some useful feedback to improve the model by designing new features.

3.2.8 Back-testing the betting strategy

Back-testing a betting strategy is a very complicated task for several reasons:

- It is hard to get reliable historical prices with an accurate time-stamp.
- It is hard to model market impact or whether a bet will be accepted or not.
- Liquidity information is not available i.e. you don't know how much is available to bet on at the advertised price.
- The market is rapidly changing: new players coming in, others disappearing ...

Therefore, we decided at Kickdex to only use the market prices as a benchmark and focused on improving our probability (in terms of our tailored in-house probability score) as much as possible. Nevertheless, we still simulate betting strategies on unseen data (usually we leave the most recent couple of seasons for validation) just to get an order of magnitude of our expected return on investment and the Sharpe ratio. However, we never used these metrics as targets to optimise in our modelling process.

3.2.9 Deploying the model in production

All our models were deployed on Amazon Web Services (AWS) and we spent a lot of energy working with the IT team to make transition from prototyping to testing and from testing to production as smooth as possible. We leveraged tools such as `conda` and `docker` and made sure all the prototyping work was made already in a copy of the production environment to avoid dealing with version clashes or any other dependency issues.

3.2.10 Monitoring trading performance

We developed automated tools to monitor the trading performance of our production model. In particular, we tried to check if the live performance was in line with the historical performance and to detect any change point in the feature distribution, the probability score distribution .... These tools allowed us to act quickly when an issue was detected and hence to avoid unexpected losses that could dramatically damage the return of our strategy.

3.3 Consultancy

Driven by the desire to apply my research to real life industrial problems, I have always been keen to collaborate with the industry during my academic career. Below are some examples of collaborations I recently did:

- Between 2008 and 2010, I worked with TOTAL to define systematic tests to one of their software (STAGE) responsible for the optimisation of the number of ships required to ensure a smooth transportation of LNG between loading and unloading terminals across the world. I helped discovering some bugs and suggested some optimisation to version 6.1.10.
- My research paper [Heil et al., 2012] found a real life application at THALES.

- In 2013, together with two colleagues from the maths department at the University of Manchester, we helped bet365 fix some bias in their pre-match pricing model for lower tier football leagues in Europe.
- I helped the UK atomic weapons establishment in 2014 to solve non-Gaussian dynamic regression problems and implemented an R package for them to apply it on their (confidential) data.
- An optimised version of the model described in [Boshnakov et al., 2017] was adapted to build some learners used by the ensemble model we deployed in production at Kickdex since 2015.
- I co-designed the sports analytics Machine (SAM) used by the BBC from 2015 until 2018 to analyse football matches, rate players and estimate the likely impact of new signings in football.
- A lawyer house used some of the football player ratings system I created [Guiliang et al., 2020, Kharrat et al., 2020] in 2018 to build a case against a football club on the behalf of a young professional player claiming some compensation for a long term injury suffered during a game against that club.
- I helped L'Oreal in 2019 to improve their customer suggestion tool by reviewing their modelling process and optimising its predictive power which substantially increased their conversion rate (visit to purchase) by 13%.

4 Toolbox

4.1 Machine learning (ML) and Artificial Intelligence (AI)

- I have a good understanding of the main ML models for supervised (classification, regression) and unsupervised (clustering) learning. I have experience with the most popular algorithms and I have been using them regularly over the past 6 years.
- Solid experience in deep learning, its main framework `tensorflow/keras` and `pytorch/fastai`, and its applications (computer vision, natural language processing, tabular data ...).
- Good knowledge of (deep) reinforcement learning. See also [Guiliang et al., 2020].

4.2 Programming languages

4.2.1 R

R has become my go to programming language whenever I want to code something. I have been using it almost daily over the past 10 years and I managed to build a powerful toolbox that would allow me to do most data-related tasks. I am the author or co-author of more than 50 packages (just a couple of them available on the public domain).

Some of the tasks I did in R are briefly described below:

- Packages implementing some predictive models based on the renewal counting process or adaptation of some survival models.
- I created `RESTful` API using the `plumber` package to expose some of my models.
- I created more than six scrapers (all running in production) using the `rvest`, `httr` or `Rselenium` ecosystem.
- I extended the `mongoDB` R driver (`mongolite`) in my `mongoTools` package.
- I am very familiar with the machine learning ecosystem in R namely `mlr` (and its successor `mlr3`) and `caret` (and its successor `tidymodels`).
- Working experience with the time series tools (`forecast`, `KFAS`, ..)
- Solid experience bridging R with other languages such as python (`reticulate`, `rpy2`) and C++ (`Rcpp`)
- Good working experience with deep learning frameworks such as `tensorflow` or `keras` in R (although for deep learning I prefer using python).
- Some experience with the main visualisation tools such `ggplot2` and `shiny`.

4.2.2 python

Most of our code at Kickdex is written in python and hence I contributed to several internal modules. I am not as advanced in Python as I am in R but I have developed:

- Good understanding of core python library and fundamental concepts.

- Some working experience with data science tools such as `numpy`, `pandas` and `scikit-learn`.
- Solid experience with deep learning frameworks such as `tensorflow/keras` and `pytorch/fastai`.
- I am able to write production ready code but maybe not as fast as I would be in R.

4.2.3 C++

The first programming language I learned is C/C++. I was using it regularly between 2008 and 2011 but slowly switched to scripting languages over the past five years. Some of the projects I did in C++ are:

- Most of the code I wrote when working for *Societe Generale* was in C++.
- I implemented the Helmholtz module in oomph-lib [Heil and Hazel, 2006].
- My R package Countr [Kharrat et al., 2019] is mostly written in C++ and then bridged to R using `Rcpp` [Eddelbuettel and Balamuta, 2017].

4.2.4 julia

I am a big fan and early adopter of julia and I started using it in prototyping work. However, due to its short history, I didn't use it in production yet.

4.3 Database technology

I have more than 5 years experience with `mongoDB` in production. However, I have only limited experience with other type of database technology such as `SQL` or `Casandra` ...

4.4 Operating System

I have very good working knowledge (more than 8 years) with Unix type operating systems (mainly Ubuntu and `redhat` Scientific Linux).

4.5 Cloud computing and model deployment

- I have been writing production ready code over the past 4 years mainly in R and Python.
- Good knowledge of version control tools (`git`).
- Good experience with dependency management system such as `conda` and `docker`.
- Some experience training complex models in the cloud (mainly Google cloud and to some extent AWS) or on internal clusters.

References

[Baker and Kharrat, 2018] Baker, R. and Kharrat, T. (2018). Event count distributions from renewal processes: fast computation of probabilities. *IMA Journal of Management Mathematics*, 29(4):415–433.

[Biecek, 2018] Biecek, P. (2018). Dalex: Explainers for complex predictive models in r. *Journal of Machine Learning Research*, 19(84):1–5.

[Boshnakov et al., 2017] Boshnakov, G., Kharrat, T., and McHale, I. G. (2017). A bivariate weibull count model for forecasting association football scores. *International Journal of Forecasting*, 33(2):458–466.

[Caruana et al., 2004] Caruana, R., Niculescu-Mizil, A., Crew, G., and Ksikes, A. (2004). Ensemble selection from libraries of models. In *Twenty-first international conference on Machine learning - ICML '04*, page nil.

[Eddelbuettel and Balamuta, 2017] Eddelbuettel, D. and Balamuta, J. J. (2017). Extending extitR with extitC++: A Brief Introduction to extitRcpp. *PeerJ Preprints*, 5:e3188v1.

[Guiliang et al., 2020] Guiliang, L., Yudong, L., Oliver, S., and Tarak, K. (2020). Deep soccer analytics: Learning an action-value function forevaluating soccer players. *Data Mining and Knowledge Discovery*, nil(nil):nil.

[Heil and Hazel, 2006] Heil, M. and Hazel, A. L. (2006). oomph-lib–an object-oriented multi-physics finite-element library. In *Fluid-structure interaction*, pages 19–49. Springer.

[Heil et al., 2012] Heil, M., Kharrat, T., Cotterill, P. A., and Abrahams, I. D. (2012). Quasi-resonances in sound-insulating coatings. *Journal of Sound and Vibration*, 331(21):4774–4784.

[Johnstone et al., 2011] Johnstone, D. J., Jose, V. R. R., and Winkler, R. L. (2011). Tailored scoring rules for probabilities. *Decision Analysis*, 8(4):256–268.

[Kharrat et al., 2019] Kharrat, T., Boshnakov, G. N., McHale, I., and Baker, R. (2019). Flexible regression models for count data based on renewal processes: The countr package. *Journal of Statistical Software*, 90(13):1–35.

[Kharrat et al., 2020] Kharrat, T., McHale, I. G., and Peña, J. L. (2020). Plus–minus player ratings for soccer. *European Journal of Operational Research*, 283(2):726–736.