University of Manchester

School of Computer Science
Project Report 2014

Using Machine Learning to Predict Personal Expenditure

Author:
Pez Cuckow

Supervisor:
Dr Gavin Brown

Abstract

A complete understanding of personal finances is becoming increasingly important as the average persons disposable income has decreased due to a changing financial climate.

The aim of this project is build to an application that makes it easier to manage a users personal finances. This is split into two halves, accessing historical information in an easy to understand way and using machine learning techniques to predict future financial information . The security considerations of storing personal finance information is also considered.

This begins with a review of the existing commercial personal finance applications and the current techniques used to forecast time-boxed financial data, such as the value of a stock on the stock market, before detailing the design and implementation of the application.

Having completed the application, the performance the selected techniques are outlined, before discussing possible extensions to the application to improve it's accuracy and increase it's feature set and possible further research.

Still needs adjusting

This needs adjusting

Project Title: Using Machine Learning to Predict Personal Expenditure
Author: Pez Cuckow
Degree: Computer Science with Business and Management
Supervisor: Dr Gavin Brown

Keywords: Markov Chain Models, Weighted Arithmetic Mean, Responsive Web Design, Web System Security

Contents

List of Figures

List of Tables

Glossary

category transactors have a category and a subcategory, e.g. Tesco = Shopping, Groceries. 13, 14, 19

global transactor the system holds two collections of transactors and mapping's, the global ones are shared between all users, and only accessed with the admin panel.. 22, 39

mapping this connects the reference found on a statement to a Transactor. e.g. Snbs, Sains =¿ Sainsbury's. 22

reference the memo or message that is included on the bank statement with a transaction. 10, 22, 23

transaction a single movement of money from/to a Transactor. 9, 10, 66

transactor somewhere money is spent, e.g. Tesco, Sainsbury's, Byte Cafe.. 21, 38, 39

user transactor unique to each user. 22, 39

Chapter 1

Introduction

Traditionally the management of personal finances is performed by viewing bank statements provided by the users bank. In the modern age of 'Internet banking', banks offer a limited set of tools that mimic the paper statements seen historically.

This project sets out to build an online application that can be used to manage personal finances. There are two main parts of the project; firstly users can upload bank statements, which are displayed and navigated in an intuitive manner; secondly, once the application has enough historical data, predicting the users future outflow.

1.1 Motivation

There are four main steps when producing and using a budget: recording previous expenses, sorting these into categories, using this historical information to estimate future expenditure, and evaluating the accuracy of predictions based on the new information and adjusting accordingly.

Since the liquidity crisis of 2009 [1], budgets have been squeezed and the average personal disposable income has fallen significantly, hitting a nine-year low in 2012 [2]. Experts suggest that "Budgets are essential for financial planning" [3], research suggesting that personal budgets lead to a "positive impact" on "mental wellbeing" [4] and guides from UCAS, the UoM SU Advice Centre and The Manchester University Crucial Guide encouraging use of budgeting, it is clear that producing a budget is of benefit.

In an informal survey[1] by the researchers, however, the majority of students questioned did not heed this advice, and were not following a budget. Producing an easier way to manage personal finances and predict future outflow can hopefully reduce the barriers to entry for creating budgets and increase the people using one.

Increasing use of debit cards [5] means that bank statements contain more and more information about where people spend their money. With access to those bank statements now provided online, and most UK banks offering the option to export transaction history, individual users can collate a database of their personal spending habits.

[1] Appendix A

Chapter 5

Results

5.1 System Walkthrough

Below, an overview of the systems behaviour is given following the standard users use case in the form of a walk through. For further exploration of features the application can be found online at `https://secure.pezcuckow.com/` where an account can be registered for access.

Having logged in, the application starts on a welcome screen (Fig. 5.1) which fills with information over time, for now the system prompts the user to upload a bank statement. The user bar appears at the top of every page[1] which identifies the currently logged in user by name and a avatar pulled using from Gravatar[63], it provides quick access to the user settings and allows the user to logout.

On the left hand side of the screen is the main menu (Fig. 5.2), which is also displayed on all pages. It indicated the currently open page in orange, in theme with the rest of the website, and on hover with a mouse it highlights the chosen page by inverting the colours to make the selection clear.

[1]For clarity both the background and bar are not shown in the following screenshots

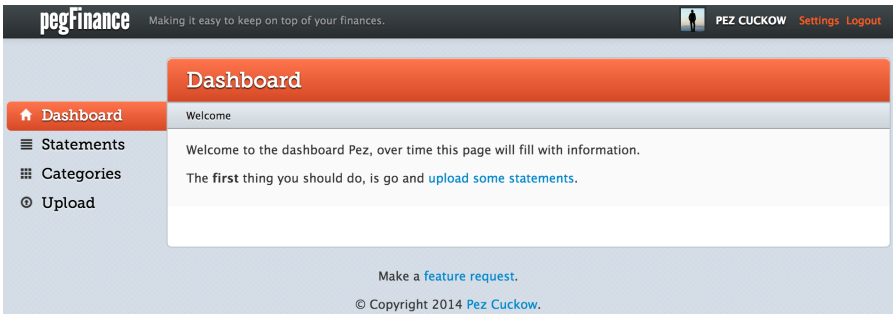


Figure 5.1: Empty welcome screen

Figure 5.2: Application main menu

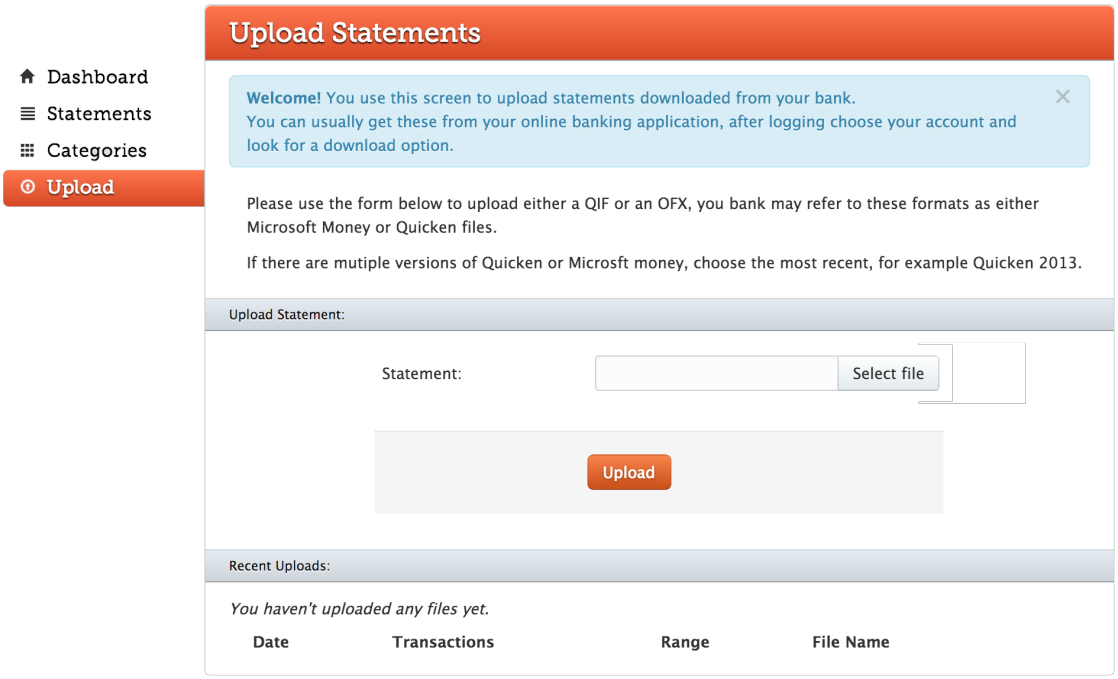


Figure 5.3: The upload statement screen before any uploads

5.1.1 Statement Upload

Following the advice of the welcome page, the user opens the upload screen (Fig. 5.3). As this is the first time user has opened this screen an information prompt is shown which explains the purpose of the page and reminds them that the files to upload can usually be found on their current banks Internet banking system.

After downloading a statement from their account, the user uploads it by clicking on the select file button and browsing their computer for the file. After confirming their selection the state of the file field changes (Fig. 5.4) to allow checking of the uploaded file and to make it clear one is currently selected. Clicking the primary[2] upload button uploads the file to the server, which processes it, and refreshes the page.

On the new page (Fig. 5.5), the state of the file upload is indicated above the form,

[2]Indicated in orange

Figure 5.4: UI update following a file selection

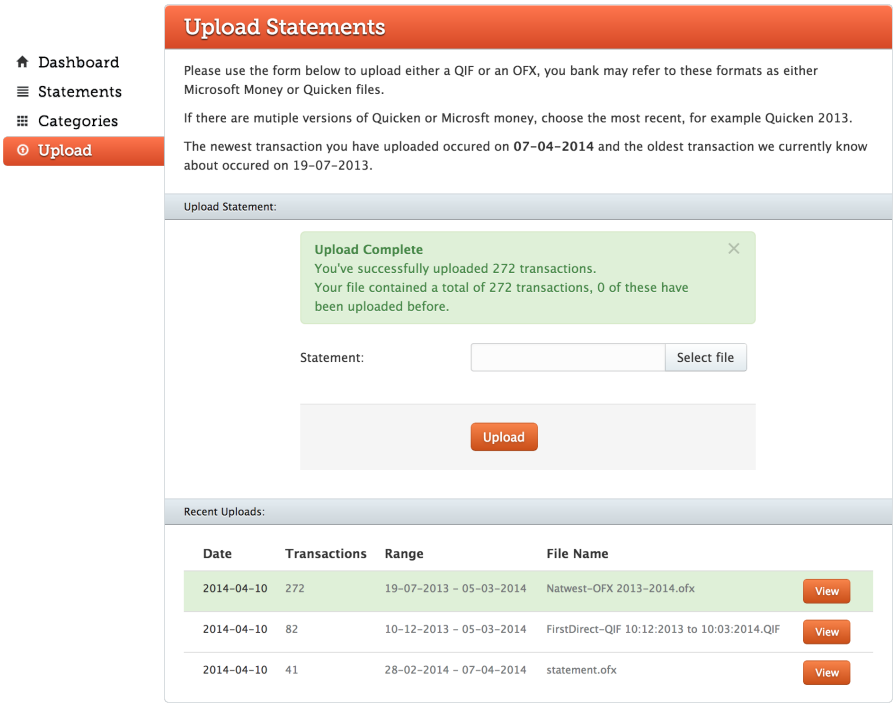


Figure 5.5: The upload page, following successful file uploads

Figure 5.6: Upload confirmation following a duplicate file

containing a summary of the file uploaded and the uploaded file is highlighted in the recent uploads list, which includes an option to view the transactions uploaded as part of that statement. The highlight colour, green, was used to indicate success.

In addition a further piece of information has been added to the page introduction, the most recent transaction the system knows about is marked in bold, to save the user time when selecting a statement to download.

If the user uploads a file containing no new transactions (all previously uploaded), the confirmation prompt indicates this and it uses the colour yellow to indicate a warning (Fig. 5.6).

On returning to the welcome screen, the dashboard is now full of information[3] (Fig. 5.7). It provides an overview of the average income and outgoings per month, along with an indication of the users 'profit' since the first transaction the application knows about.

Most notably, the dashboard also includes an interactive pie chart, which gives an indication as to which categories most of their money is spent. On hover the chart gives the actual percentage of the overall expenditure, and on click opens a page which lists all the transactions in that category.

If the system has suggestions for uncategorised transactors, a notification suggesting they visit the category (or suggestion) wizard is shown.

5.2 Suggestion Wizard

The suggestion wizard steps the user though all unmapped and uncategorised references (Fig 5.8). It starts with the transactor they frequent the most often and prompts them to provide the mapping, either by following a suggestion, creating a new transactor or manually searching for a match.

Completing the mapping, forwards the user to the next unmapped reference, providing confirmation the new mapping has been created as a notification which disappears after a few seconds 5.9, and the process repeats.

The suggestions wizard went through several iterations of user interface enhancements, designed to make it easier to use. For example, when creating a new transactor (Fig. 5.10) the system automatically fills in the name field with a pre-formatted version of the reference and the category selection is performed through an upgraded dropdown menu. The dropdown has been upgraded from the standard dropdown found on many

[3]Fictional information used throughout out this walkthrough

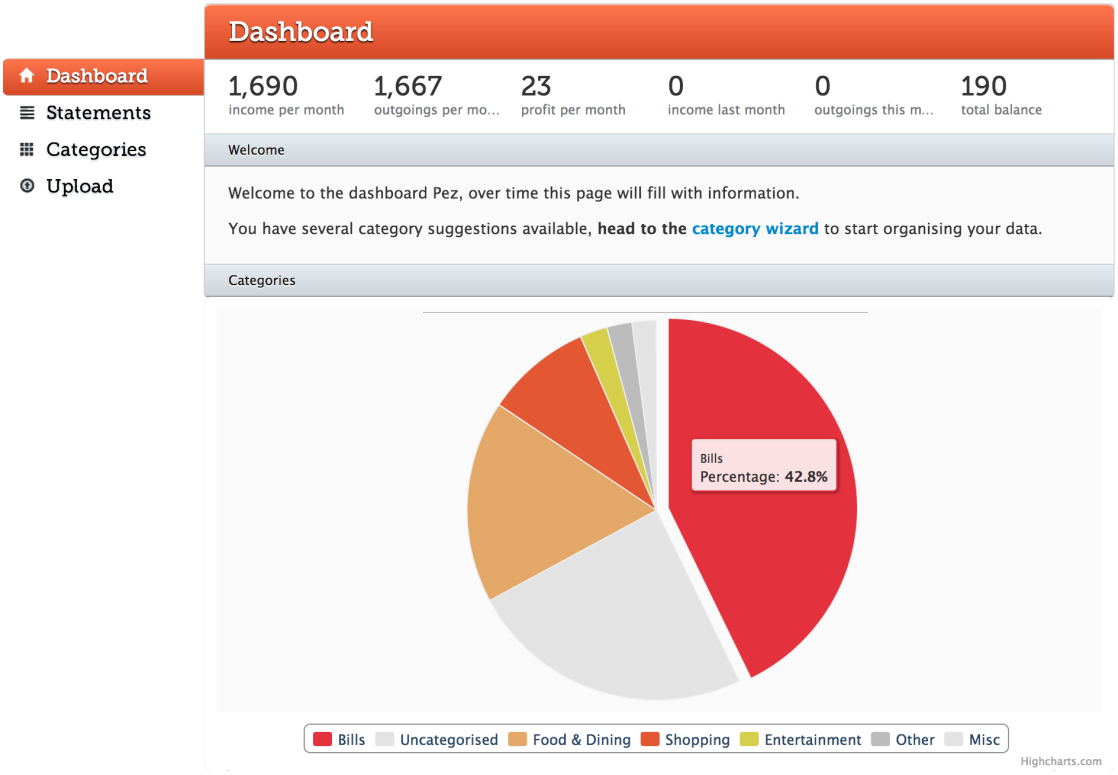


Figure 5.7: Welcome screen following statement uploads, including expenditure per category

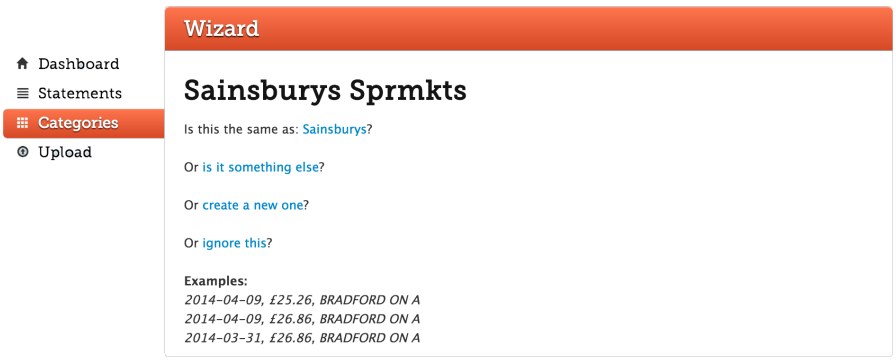


Figure 5.8: Suggestion wizard main screen, showing suggestions for a reference

Figure 5.9: Notifications shown after completing a successfully

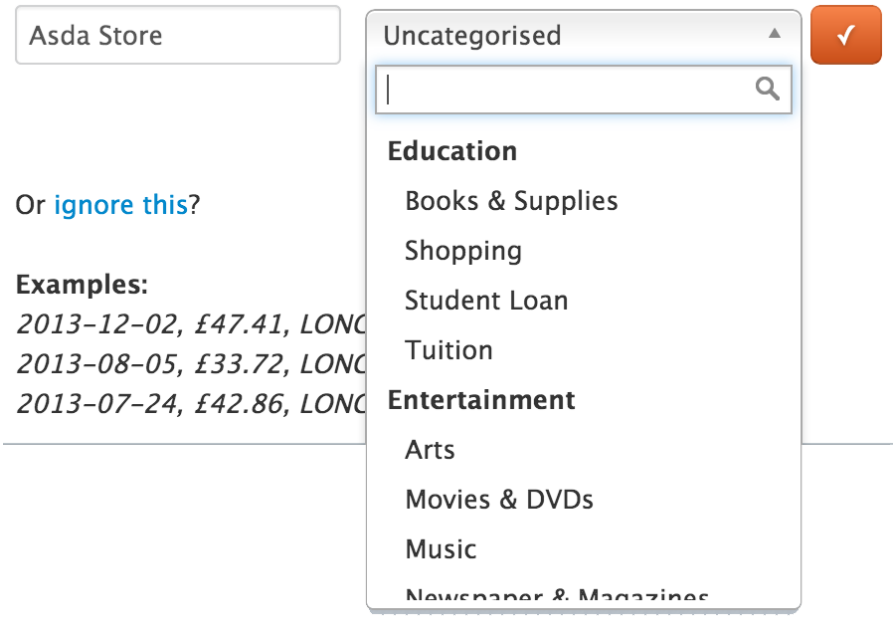


Figure 5.10: Creating a new Transactor and selecting a category

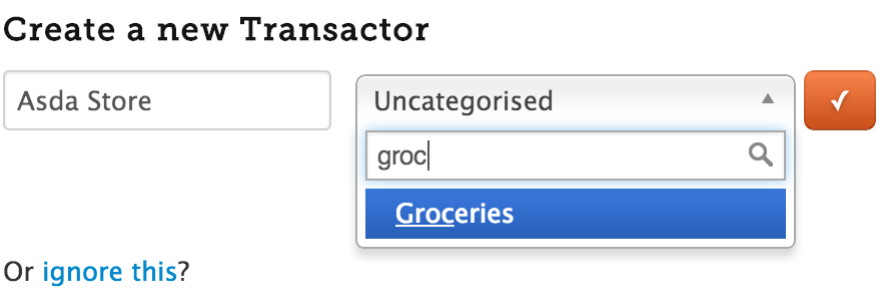


Figure 5.11: Selecting a category using the autocomplete feature

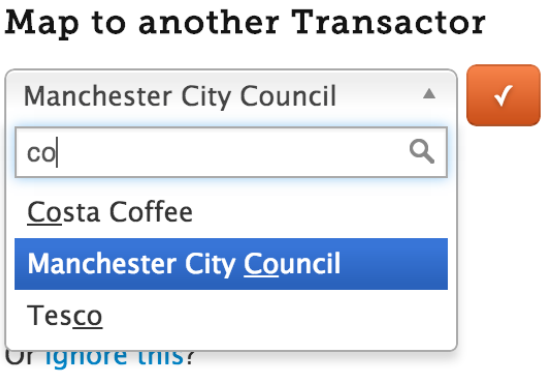


Figure 5.12: Searching for an existing Transactor using autocomplete

websites using client side javascript and provides auto-completion, fuzzy matching, as well as the ability to select a category grouping, for example entertainment over Movies & DVD's (Fig. 5.11. A similar dropdown is also used when searching for an existing transactor (Fig. 5.12.

After completing the wizard by mapping or ignoring all the unmapped references, the user is congratulated and a hint is given that they should head to the transaction summary page, which includes their monthly expenditure predictions.

5.3 Transaction Overview

The transaction overview screen shows spending per month in each of the categories and a the prediction for next month made using the machine learning techniques outlined in section 3.2 (Fig. 5.13). At the top of the screen a summary of the total income, expenditure and net profit for each month is shown. The rest of the page is split into two sections, representing money coming into and leaving the users account. In the example shown the user hasn't mapped all of their references and so some transactions are listed under uncategorised, a hint is placed at the top of the screen reminder users to visit the category wizard.

.png

Clicking on any of the rows in the table reveals the subcategories and their associated values that are being used to produce the row, using an animated 'slide-down' effect 5.14.

5.4 Viewing Statements

The application also provides an easy way to view historical spending organised by month (Fig 5.15). The transactions in each month can be grouped by category, transactor or date to give a more detailed idea of how money is being spent (Figs. 5.16 and 5.17).

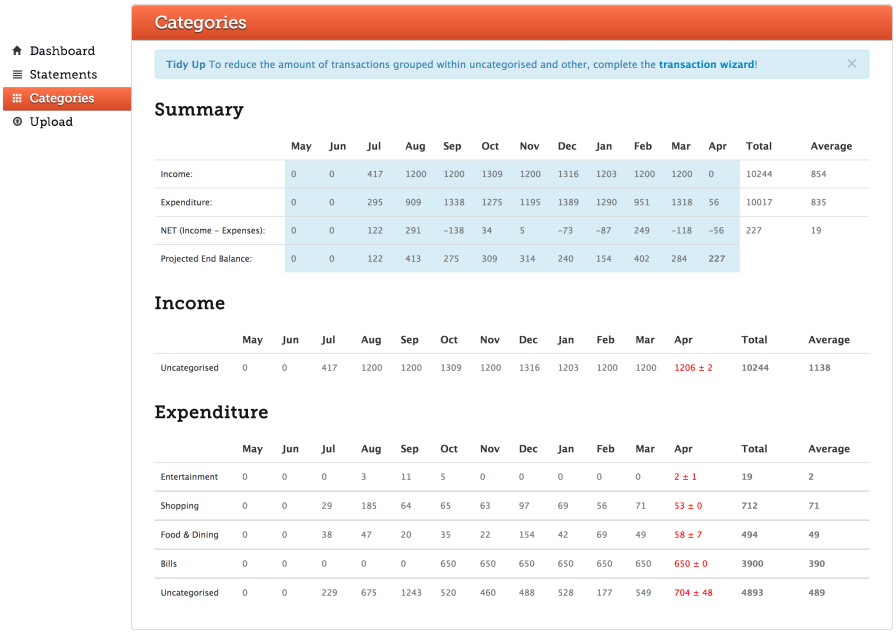


Figure 5.13: The transaction overview screen, which shows expenditure per month and a prediction (in red) for next month

Shopping	0	0	136	35	130	74	54	287	452	35	5	49 ± 8	1207	121
> Clothing			0	0	0	60	0	15	18	0	0	8 ± 5	93	9
> Hobbies			0	0	10	0	5	25	0	0	0	2 ± 5	39	4
> Uncategorised			136	35	120	15	49	246	434	35	5	37 ± 13	1074	107

Figure 5.14: The subcategories being used to make up a category in the overview

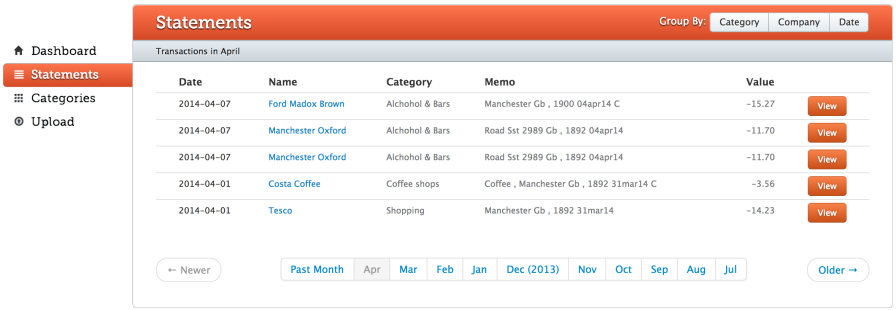
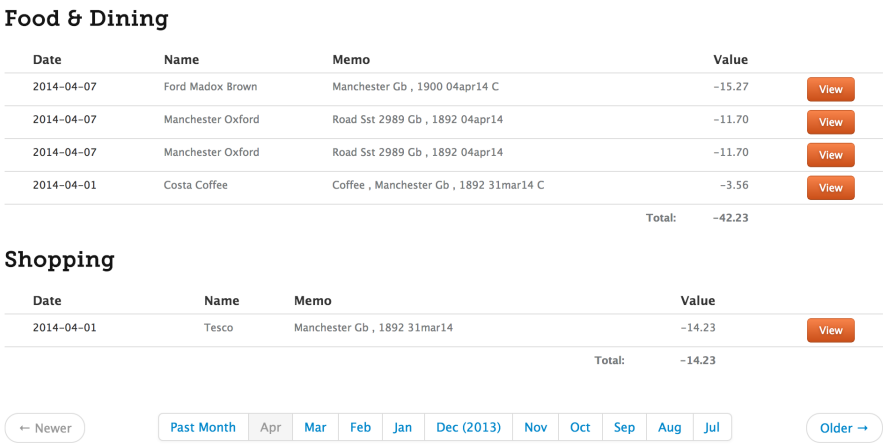


Figure 5.15: The statement view



Food & Dining

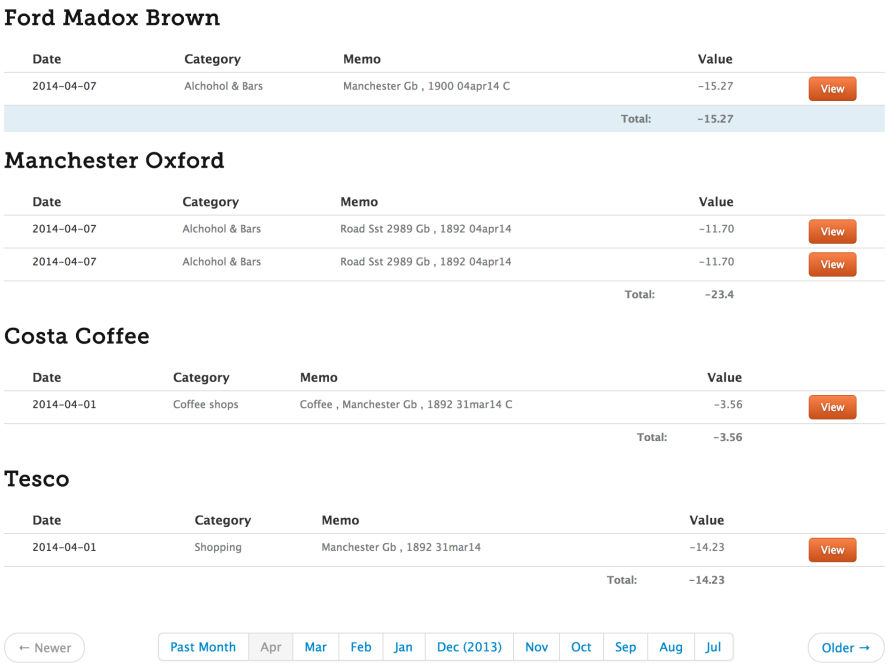
Date	Name	Memo	Value	
2014-04-07	Ford Madox Brown	Manchester Gb , 1900 04apr14 C	-15.27	View
2014-04-07	Manchester Oxford	Road Sst 2989 Gb , 1892 04apr14	-11.70	View
2014-04-07	Manchester Oxford	Road Sst 2989 Gb , 1892 04apr14	-11.70	View
2014-04-01	Costa Coffee	Coffee , Manchester Gb , 1892 31mar14 C	-3.56	View
		Total:	-42.23	

Shopping

Date	Name	Memo	Value	
2014-04-01	Tesco	Manchester Gb , 1892 31mar14	-14.23	View
		Total:	-14.23	

← Newer | Past Month | Apr | Mar | Feb | Jan | Dec (2013) | Nov | Oct | Sep | Aug | Jul | Older →

Figure 5.16: Grouping the statement by category



Ford Madox Brown

Date	Category	Memo	Value	
2014-04-07	Alchohol & Bars	Manchester Gb , 1900 04apr14 C	-15.27	View
		Total:	-15.27	

Manchester Oxford

Date	Category	Memo	Value	
2014-04-07	Alchohol & Bars	Road Sst 2989 Gb , 1892 04apr14	-11.70	View
2014-04-07	Alchohol & Bars	Road Sst 2989 Gb , 1892 04apr14	-11.70	View
		Total:	-23.4	

Costa Coffee

Date	Category	Memo	Value	
2014-04-01	Coffee shops	Coffee , Manchester Gb , 1892 31mar14 C	-3.56	View
		Total:	-3.56	

Tesco

Date	Category	Memo	Value	
2014-04-01	Shopping	Manchester Gb , 1892 31mar14	-14.23	View
		Total:	-14.23	

← Newer | Past Month | Apr | Mar | Feb | Jan | Dec (2013) | Nov | Oct | Sep | Aug | Jul | Older →

Figure 5.17: Grouping the statement by category

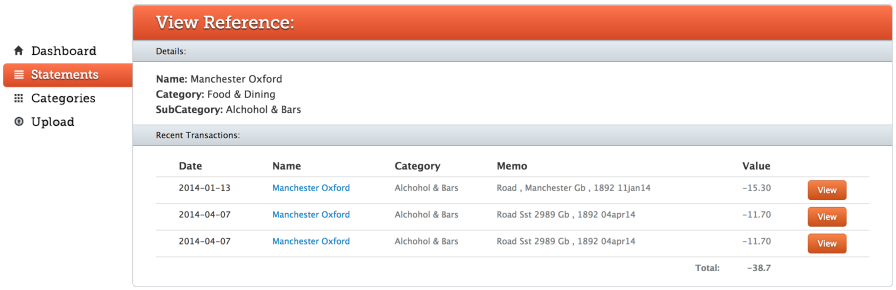


Figure 5.18: Recent transactions at a particular transactor

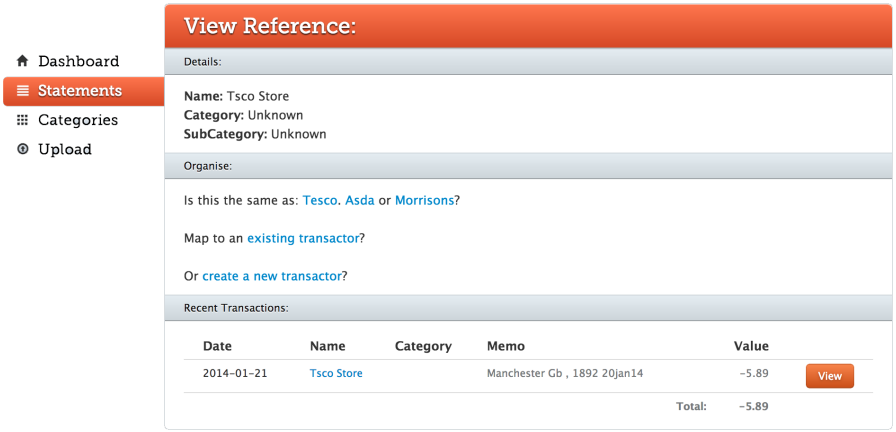


Figure 5.19: Viewing an unmapped reference

A user can also view particular transactor or reference, which includes the references category and a summary of recent transactions (Fig. 5.18). If the reference is not yet mapped, options similar to those shown in the suggestions wizard are listed enabling them to map the reference correctly (Fig. 5.19).

5.5 Responsive Web Design

A key feature of the application is being able to access it at any time from any device, particularly when taking into account the rapid increase in the use of mobile devices. Interacting with a website on a smartphone or tablet is not the same as interacting using a computer, due to the smaller screen size and use of touch over a mouse.

Forbes reported that 24% of their 2013 website visits came from mobiles and 13% from tablets, down from a total of 15% in 2012. particularly with the high percentage of website visits coming from mobile devices [64].

In order to ensure the project is accessible from a variety of different devices the core UI uses Responsive Web Design (RWD) to layout the website differently depending on the screen size of the device to ensure an optimal viewing experience.

The differences depending on the device are highlighted in Figs. 5.20-5.23.

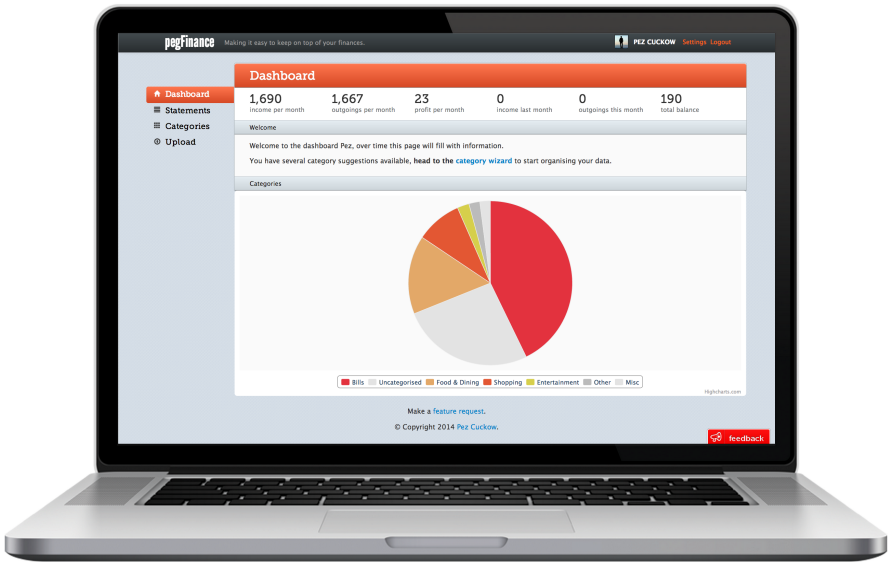


Figure 5.20: Layout on a standard laptop

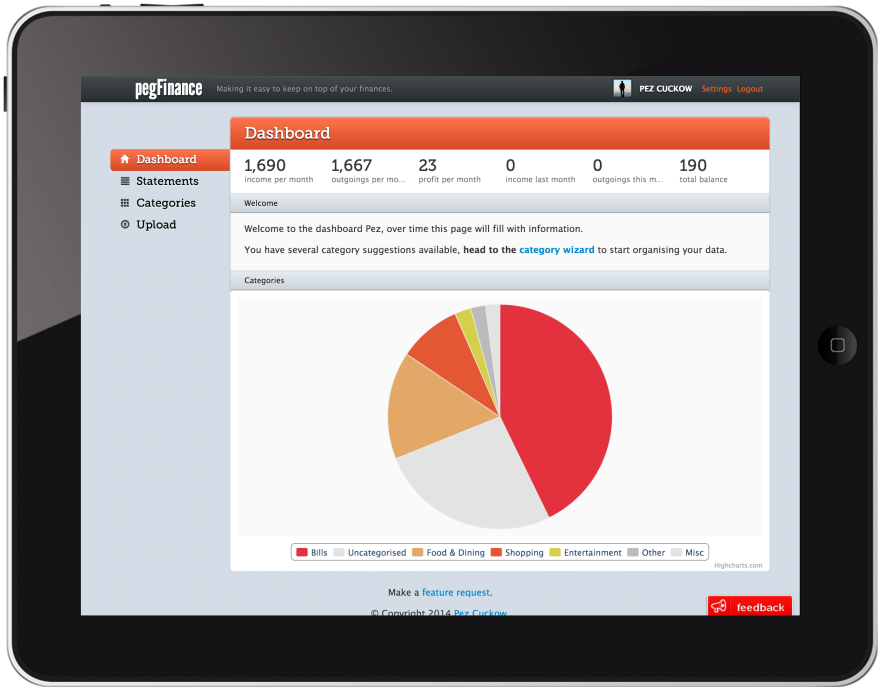


Figure 5.21: Layout on a tablet in landscape

Figure 5.22: Layout on a tablet in portrait

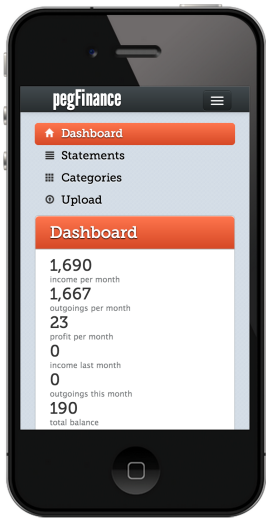


Figure 5.23: Layout on a smartphone

Chapter 6

Testing and Evaluation

The evaluation of the application indicates that these three objectives have been met. Preliminary research demonstrates that enjoyed using the application and that the majority of participants would use it in the future if implemented as a full product. Testing with participant data shows that the average absolute error of the prediction engine was $PERCENTAGE\%$ for users with over three months historical information. Preliminary penetration testing, using both white-box and black-box testing indicated that the security protections put in place are effective. However, the evaluation of the project was limited, due to the size of the user base, discussed further in section 7.1.

Complete this

6.1 During Development

Though-out the development of the project, predominantly after each feature was implemented and before a new iteration was started the applications functionality was tested through face to face conversations, observations and unit testing.

6.1.1 Acceptance Testing

As noted throughout the report, the application was regularly tested by potential end users and the developers during lab observations where the user was asked to complete a task on the website while describing what they were doing.

During the testing, common pain points were identified and these were used to decide on the future features, either to improve the user experience on the website, making it easier to use, or to support a new use case.

Example modifications, made as a result of these observations, include the dropdown menu autocompletion when creating and mapping transactors, the hints that appear throughout the website to guide the user onto the next task, and the suggestion wizard.

Towards the beginning of the project these meetings were used to evaluate and update the design of the user interface, which went through several different designs, and to receive qualitative feedback from the users (Fig. 6.1).

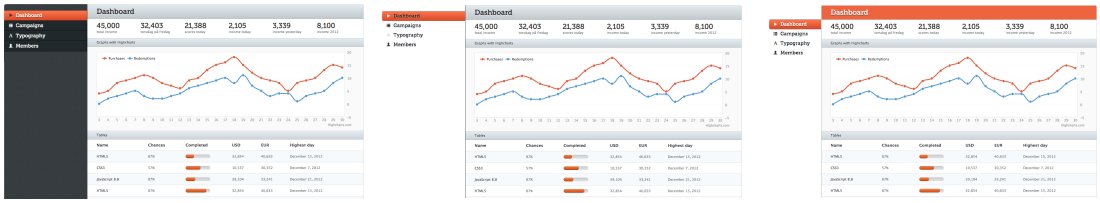


Figure 6.1: A few iterations of the user interface designs

6.1.2 Unit Testing

High risk classes, including the Budget, the QIF and OFX Parsers implementing the TransactionFileParserInterface, and those implementing the TransactionCollectionInterface were unit tested to ensure the functionality of the classes was not affected when implementing new features or refactoring the code.

These classes were selected as the core functionality of the application relied on their behaviour and they were coupled to the most other classes, for example the Budget object was responsible for holding a collection of Transactons in the tree data structure described in section 4.3 and performed the majority of the prediction functionality, holding references to the TransactionMarkovChain, TransactionWeightedAverageCalculator and PredictionEvaluation.

Due to time constraints and the complications of testing data driven classes, both the number of unit tested classes and methods covered was limited, this is discussed further in section 7.1.

6.2 After Development

Once development of the project was halted the application was tested in the three following the three sections outlined through the report. The statement management functionality was assessed using a questionnaire posed to research participants that were given access to the system, the prediction algorithms were evaluated using the sample data provided by the research participants and the security of the application was evaluated using penetration testing. Unfortunately the reliability of the results for both the statement management and prediction is questionable due to the limited number of research participants and questionnaire responses, see section 7.1.

6.2.1 Statement Management

The questionnaire posed to the research participants was split into three sections, a full copy of the questions can be found in Appendix G.

The first made a series of statements and asked the respondent to answer on a five point scale, ranging from 1, strongly disagree to 5, strongly agree. The questions in this sections had four main types, and were selected following the structure of the Standardized Universal Percentile Rank-Questionnaire **??** which is used by companies including

PayPal to assess their websites. Questions on usability focused on how easy users found navigation of the website, locating the information they needed, and whether they enjoyed using it. Credibility questions focused on whether the user trusted the website having uploaded their personal information. The likely hood of a user recommending the site to their colleagues and friends or returning to the site in the future gave an idea of the potential loyalty the participants expressed. The last focused on the user interface to gain an idea of whether the website appealed to the users.

The second section focused on task completion, participants were asked to complete a task and then rate the how difficult or easy they found that on a seven point scale. This section was designed to assess the functionality of the website and to identify additional pain points. One scale question was asked per task as research suggests that use of a single question performs as well as breaking the task down into sub-sections [65], [66], however a free text field was included for each task to obtain qualitative information on the reasons behind the answer.

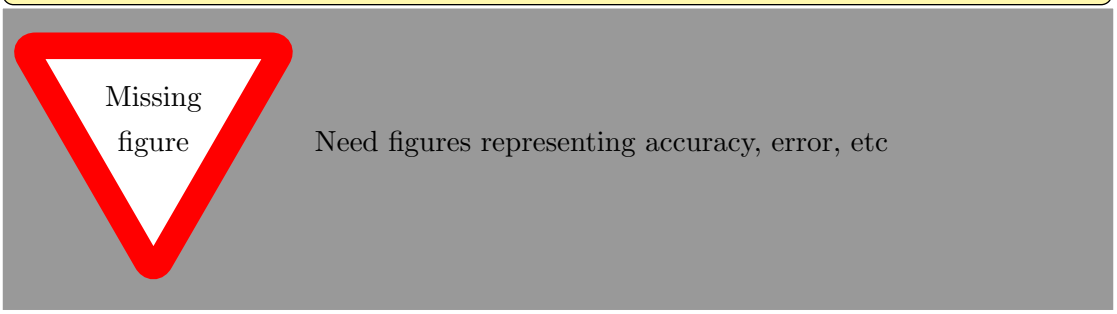
Standardized Universal Percentile Rank-Questionnaire

SEQ - post tast

6.2.2 Prediction

6.2.3 Security

P3: How did I check that it was working? Testing with personal data, users trying it and running tests of everything



6.3 User feedback

P4: User surveys to check what they wanted

Chapter 7

Conclusions

The project set out to build an application that made personal finances easier to manage through the automation of the common steps that people go through when making a budget.

Existing finance applications were researched, highlighting the advantages and disadvantages of each, as well as identifying the features that users found useful, through use of reviews. A key piece of the functionality, the prediction engine, was discussed, investigating a combination of existing techniques used to forecast financial spending, including MCMs and Weighted Arithmetic Means. The ethical implications of storing high risk personally identifiable information was assessed and considered with reference to application security and strong security practice. This research was then used decide on the functionality of the application, the design and implementation of the prediction engine and used to ensure high security standards were met throughout the application.

Specific design and implementation challenges were discussed, highlighting the key background and technical knowledge that was gained in order to overcome the challenges faced.

Fix this

The aim of the project was to build an application that implemented an intuitive way to view and manage personal finances, accurately predict a users future transactions based on their history, as well as upholding the high security expectations of such a service. The final application was reviewed following the usage pattern of the average user, to outline the key features implemented and to give an idea of how it works.

The evaluation of the application indicates that these three objectives have been met. Preliminary research demonstrates that enjoyed using the application and that the majority of participants would use it in the future if implemented as a full product. Testing with participant data shows that the average absolute error of the prediction engine was $PERCENTAGE\%$ for users with over three months historical information. Preliminary penetration testing, using both white-box and black-box testing indicated that the security protections put in place are effective. However, the evaluation of the project was limited, due to the size of the user base, discussed further in section 7.1.

Complete this

7.1 Limitations

The main limitation of the project was the scope of the test participants. A total of 20 testers signed up to participate in the project during it's development, uploading at least one month of transaction data. However, only 7 of these individuals uploaded more than 3 months of transaction history, which was the threshold for using training and testing to select a best fit weighting model. In addition the majority of the participants were peers of the author, which meant that the majority of evaluation and testing was performed using students.

More extensive testing using a broader spectrum of testers, who may require a variety of different features, or use the system in a different way, may provide a greater level of evaluation for the system.

A key difficulty when seeking participants were their concerns regarding data security and trusting the researchers with their personal information. It it hoped that by publishing the level of concern and detail that has gone into the design of the security of the application, including encryption of their personally identifiable information, that the effects of this concern can be limited.

If the project were to continue, a more open approach for recruiting participants could be taken, such as seeking alpha testers and feedback in online forums and news groups. This should lead to both larger numbers of participants and a wider variety in the users background to give a fairer picture of potential users. Due to the variety of formatting already seen in participant data, it may be appropriate to limit the participants to the UK. A continued concern would be the security of the application when opening it up to users on the Internet, therefore this would need to be addressed.

During the testing and evaluation there were many features suggested by users to improve the application to their individual preferences. Due to time restraints the majority of these were not implemented and it was decided it was more imperative to focus on the most important and most frequently requested features first. Further development could include the additional requested features to suit a greater variety of users.

Although the key functionality of the application was unit tested, tests of the business logic in the prediction engine and on key objects was limited. This limitation was due to the heavy reliance on data, and because the engine performs differently depending on the users history. To test this functionality in the future data with pre-calculated attributes would be need to be prepared and a separate database access for storing this testing data would need to be implemented.

7.2 Further research

Due to the nature and size of this project a wide variety of further research opportunities have been highlighted.

7.2.1 Model Selection

Currently, the application selects most appropriate weighting model to use for each users transactions by evaluating the pre-defined weighting models via their historical data, and selecting the one with the least total overall error.

User Classification

Given the wealth of information the application holds about the users, higher accuracy could potentially be achieved by classifying each user. The users could be classified in two different ways; using pre-defined groups, or using cluster analysis.

To split the users into existing groups, the researchers could identify groups that they believed would exhibit similar spending patterns and characteristics. Users would identify which group they felt they fit into when signing up and this information would be used to define the characteritics of each group. Having chosen features that identify each group, the researchers could automatically classify the users without prompting them

Alternatively, the use of cluster analysis could be employed to determine the number of groups, as well as the members of groups. The objective of the clustering would be to group users that exhibited similar spending patterns together, so those that were most similar were together and more similar to those in other clusters.

Having classified the users, the parameters in the prediction engine could be configured for each individual group rather than by user. In addition, it would allow similar users to be compared, which would allow statements such as 'users similar to you spend, on average, 10% more' to be made, and could perhaps be used to make savings suggestions to each user.

Further research could investigate the reliability of the classification using different clustering algorithms and whether configuring the prediction engine per group leads to better results than per user.

Model Selection

Weighting models were currently selected by category. The users total transactions per month in a particular category were evaluated and the weighting model with the best fit is chosen for that category. Transactions were considered by category and by month. Data was split by category which was chosen as the number of transactions in each subcategory was low, and preliminary experimentation showed that sub-categories would suit the same model as their category. Using months as the time period was selected as the majority of participant data contained regular monthly transactions such as household and mobile phone bills, and so considering the transactions at smaller time steps led to errors in the Markov Chain Model as the probability of a transaction not occurring increased.

However, different time periods and methods of splitting the data could be investigated. For example transactions at each individual transactor could be assigned their own model, or all spending below a certain value could use the same model. The time

$$\bar{x} = \frac{\sum_{t=0}^{n-1} w(t) \times x_t}{\sum_{t=0}^{n-1} w(t)}$$

Figure 7.1: Weighted arithmetic mean

periods used to build the model, could be by month, weeks, days or perhaps even by day of the week. Use of a time period such as a day of the week would result in a more complected Markov Chain Model where samples such as 'given that the last purchase was on a Wednesday' could be taken.

Further research could investigate the effects of using different grouping methods and time periods on the overall accuracy of the predictions.

7.2.2 Alternative Forecasting Techniques

The system currently uses techniques adapted from exponential smoothing to forecast and predict the value of a future transaction. Currently this is limited to a weighted arithmetic mean in the form shown in Fig. 7.1. Use of a mean to forecast the value of the next period performs poorly when there is a trend in the data [21].

Second order or double exponential smoothing can be used to take into account trends and seasonal changes in data, such as the average value of an individuals weekly shop increasing over time, or the increase of purchases during Christmas. Double exponential smoothing involves two steps, calculating the weighted mean as before, and then adjusting that mean by the difference between the previous value and it's associated prediction, shown in Fig. 7.2, the comparison between the weighted mean and second order forecasting transaction that is increasing in value by 10, for each time period is shown in Fig. 7.3. Triple exponential smoothing extends the trend calculation to include a seasonal index. Assuming at least one complete seasons data is available the value of x_t of scaled by this seasonal index, which is calculated using by taking the average value of this period historically and dividing it by the average value for the year that the period occurred or $S_t = \frac{x_t}{\bar{x}_p}$ and $\bar{x}_p = \frac{\sum_{i=1}^{p} x_i}{p}$ where p is the number of periods in a year, and x_i is the value of the period i in the year considered. Fig. 7.4 shows the performance on some data that follows a similar trend to the previous year [20], [22].

Further research could investigate the effect of applying double order and triple exponential smoothing to forecast the values of transactions on the final prediction and how seasonal indexes could be calculated. Seasonal indexes, for example, could be calculated per transactor, category, user or globally across the application. The research would need access to historical transaction data for the users to be able to make these assessments.

7.2.3 Learning the Scaling Parameters

It was mentioned in section 3.2.3 that the speed of the decay in the weighting functions could adjusted by using a scaling constant. Instead of choosing the best weighting model from a pre-defined selection, the application could apply non-linear optimisation

$$\bar{x}_t = \alpha(t) \times \bar{x}_t + (1 - \alpha(t))(\bar{x}_{t-1} + b_t - 1)$$

$$b_t = \alpha(t)(\bar{x}_t - \bar{x}_{t-1}) + (1 - \alpha(t))b_{t-1}$$

Where $\alpha(t) = \frac{w(t)}{\sum_{t=0}^{n-1} w(t)}$ representing the final weight
In general, the starting values are set as $\bar{x}_1 = x_1$ and $b_1 = x_2 - x_1$

Figure 7.2: Calculating the second order weighted arithmetic mean

Figure 7.3: Comparison of first and second order forecasting when predicting a trend

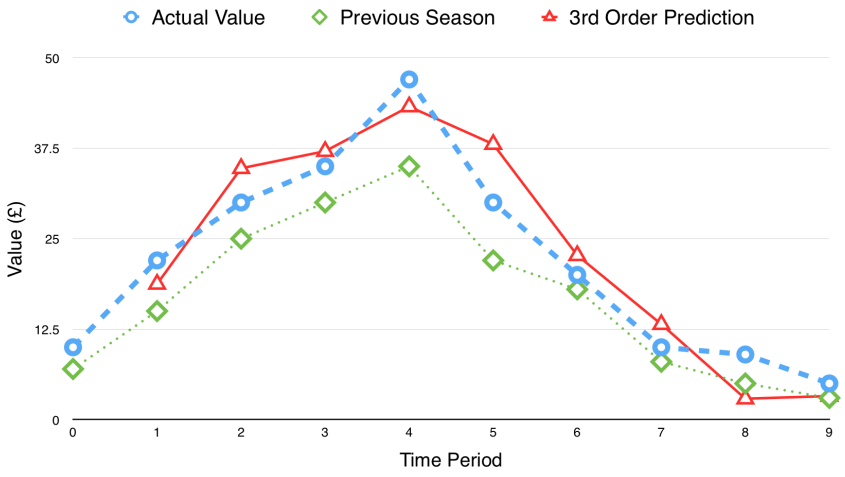


Figure 7.4: Using third order forecasting on data with a previous season

techniques to adjust the rate of the decay, and select a customised model for each user.

Further research could investigate different implementations of these optimisation techniques, and how the use of custom models affects the prediction accuracy for each user.