# Define the problem

Introduction

Our company calculates legally qualifying expenditure related to R&D projects in order to retrieve tax relief for our clients.

We operate on a success based fee structure as we will take a proportion of the final  
benefit for our clients. In the course of securing this benefit, we will proceed through an initial call, physical meeting / call, collect financial documents, expenditure information, compile a justification and submit to the tax authorities. At any stage, legal, technical or financial factors may hinder the process, or prevent success completely. We operate in a small market, it is a niche industry, and our clients will not have a clear understanding of where our services will take them, and it is likely that we won’t know either due to the complexities of the industry. We need to establish greater clarity on our process as early as possible in our process, so that both our internal staff and our clients understand the eventual benefit or cost of continuing. What would be the potential benefit to our clients who don’t know anything about this specialised industry?

# Identify your client

My primary client is my own company, to strengthen the company’s understanding of potential success or failure in a client’s claim for tax relief. With a better understanding at the forefront of our process, we can substantiate greater evidence for the returns of our product, improve our qualifications of our clients, provide a better service altogether, and improve decision making of how our clients should be handled.

# Describe your data set, and how you cleaned/wrangled it

Extraction of staff costs

Cost summaries are first copied into a folder so that the script isn’t too intensive in scanning over the entire internal drive each time the operation is completed.

Next is a for loop which loops over every worksheet in every spreadsheet. A function ‘get data’ is used to iterate over every row in each worksheet which retrieves the data into pandas Data Frames. A function called ‘staff extract generator’ is used is called, which drops empty cells and applies the squeeze nan function; this reformats the Data Frames to restructure a summarising ‘total costs’ cell to follow a heading ‘staff cost’. Between the heading and total costs in where the data lies. As these ‘labels’ are not consistently applied due to either error or process changes, the labels are modified to be uniform. Whitespace is stripped from the index as a series, and using lambda functions the labels are changed. Lambda is able to approximate the cell labels, therefore we can use this approximation as an if function, if the cell contains the pattern we want, which translates to a ‘staff’ heading or ‘total staff costs’ then it is changed to s total costs and staff costs. We can then add the edited index back into the Data Frame index and loc can be applied to specify the data required.

The filename and worksheet is then extracted while the spreadsheets are being manipulated, which will be utilized as a reference point, allowing us to locate the associated client in our main systems.

The data we require is only in the first four columns of the data set, and the spreadsheets may contain notes which would corrupt this data set, therefore the first four rows and created reference points are extracted from each worksheet.

As the spreadsheets we are iterating over contain worksheets that do not include the data set we require, a try and except function is applied, and the script prints the exception in case there are any worksheets that do contain our desired data that are not extracted due to errors.

Cleaning

Next is the ‘concatenated df’ function which creates a single Data Frame from each worksheet Data Frame. We can then begin to give shape to the final dataset. The column names are titled, including ‘r&d cost’ which is included in the original set but not included in the final Data Frame; but it is included at this point in case, although unlikely, a spreadsheet included additional information, i.e. the jo role and project role.

A reindex is used so that we can refine the job titles used in the staff costs. A regex is applied which identifies entries in the job titles column with text only, as columns which would have numeric values would class as an entry without a job title or staff name. At earlier stages of our company’s process in treating these costs we have not included the staff member’s name, and in this process these entries are eliminated. Although they are entries that contain apportionments and costs they don’t include the job role, which is vital, and I would consider this data less reliable. A similar validation is used against the apportionment and total cost. A lambda is applied to both columns which ensures that they are int or float types only. Strings are not desired. As the entries might be entered incorrectly, MAX and MIN functions are applied to extract the cost and apportionment respectively, under the assumption that the greatest figure will be the total cost, the second figure will be the calculation between the total cost and apportionment (not required) and the apportionment is the lower figure. The apportionment must also be between 0.1 and 100 to be considered a percentage of total costs. If there are any nan cells after this process, then the cells are dropped.

Extracting identifiers to connect to client data

With the data set correctly formatted, and the bulk of data in correct order, the cif (ID) of the correlating client can be associated with the database of client data. The cif is not always in the filename, therefore the best way to extract the cif is from the data source, from our internal servers using ‘client id’ function. A Data Frame is constructed with file destinations including costs summary or cost summary as the data was originally extracted. A merge is completed between the Data Frame of file destinations and the staff costs dataset based on filenames. To ensure that the original index is not lost during the merge operation, a reindex is used and the job titles is ultimately used as the index. Based on the foldernames extracted merged into the Data Frame, a regex is utilized to extract the correlating cif number. As many clients have three to five-digit numbers in their company names, the regex excludes these recognised digits in the Data Frame. File names might also include the correlating periods including 2013-2017 or expressed as 14-17, therefore these periods have also been excluded. If more than one cif has been extracted from these file destinations, then the function ‘tidy cif’ is used to split the digits extracted into separate rows. The cif column is then cleaned to remove characters created by the lists and remove empty cells, and their associated rows.

The year is then added from the worksheet of each spreadsheet. This might include the periods between certain dates or with a year, therefore a regex is used to identify the last two digits before the “] characters used in entering the worksheet names, and the ultimate two digits from this created dataset is extracted, ensuring that if two dates are extracted then the final two digits are the digits that represent the year-ending period. At each stage the created series’ have a function applied to remove the list formatting to return only the two digits we require.

Final cleaning

The remaining step in cleaning the data is to remove duplicates, which is applied by subsetting the names, cif and worksheet, assuming there are no two individuals with the same name in the same period acting on behalf of our clients. The columns we need for the final dataset are selected, the cost, percentage, cif and year. With the cost DataFrame obtained, it can be connected to an Excel spreadsheet which includes the client data.

Other cost brackets

The subcontractor databases is completed with key differences to the staff costs. The labels used to define the location of the data required for this data set is set to xsubcontractor and xtotal sub to create unique identifiers for the data. The columns are not selected until after the worksheets are concatenated and an additional column is added. The additional column in needed as the subcontractors have an additional column to detail whether they’re connected to the client or not – if they’re not connected only 65% of their costs are claimable. Consequently, this dataset is more vulnerable to shifted columns, where a column will have a part of another column’s data. Due to this vulnerability in the data set, the max and min functions used to unscramble the data is more important. The consumable and software data sets are created in the same manner as the subcontraction costs to ensure consistency in the code, but with corresponding labels to extract relevant data.

# List other potential data sets you could use

I could also use the report, the justification of the tax relief that our clients are due. I could use a bag of words model on this, although it probably won’t yield too great of results considering that the industry has not a significant correlation with the data. I wouldn’t be able to use this per industry or per type of project at present.

# Explain your initial findings

Costs brackets

The most important variables were to explore the expenditure itself which construct the total costs. There is significant variation in the size of each cost type; staff costs are significantly greater than software expenditure. The difference in size of data sets might reflect why the staff correlation coefficient would be much greater than subcontractor, material and software by a large margin. However we have tested each cost type where they are present, therefore this should not disrupt the data too much. It also makes sense from a project perspective that on the whole, the bulk of expenditure would be staff and the time they’ve spent on the projects, while the materials they use would not be so significant.

There is also a lot of variance within the data, with mostly small figures and large ‘outliers’, the data is exponentially distributed. It would seem that the majority of our clients would be completing ‘side projects’ and a few larger companies are generating a lot of expenditure. The distribution of each cost total vs number of line of costs indicates that it could be a good predictor that the more of each cost there is, the greater the cost, although this isn’t the case with the material and software costs, with some low correlation coefficients. The material and software costs actually seem to share the same distribution which was proven to be statistically significant. They also have similar behaviours, a similar allocation to R&D towards 100%, and roughly the same number of material/software per client. This leads to discussion of how well the characteristics of the expenditure could be used to predict total expenditure.

How much was apportioned from total staff costs for example, and how well does this translate to total expenditure? Using the median as the data is exponentially distributed, this metric has not performed well in predicting total expenditure. Staff apportionments had the strongest correlation but it was still very weak. Although on its own it does not perform well, I would still surmise that it would be useful along with the number of staff or payroll to estimate total expenditure. Another characteristic of the costs would simply be counting each cost in each bracket per client and year. Again it is the staff members that have the strongest correlation. The other three cost brackets have moderate correlation to the total costs in this method. The count certainly should be an useful contribution to the prediction.

Client data

Industry was a feature that I expected would be significant to predicting R&D value – I would expect that a client in the mechanical engineering industry would have much greater R&D expenditure than a client in the food and drink industry, but the difference is minimal, and there is very little correlation. Although some industries are much greater than others in terms of R&D, and certainly in the number of our clients per industry, each claim for tax relief cannot be differentiated by industry. Postcodes was another feature I had expected to be useful to predict R&D expenditure. This had a very low correlation score also, and was not useful to predict expenditure. The postcode in particular shows the weaknesses of the dataset in how few records there are per post code. Of the top 15 most frequent post codes, there is only 643 records. There is indeed some regions with very high values but the data was not statistically significant. The final variable which I had high expectations for was the phase, the number of times a client has worked with us. Judging from the mean of each phase, the more times a client works with us the greater the value of the benefit, however from looking at the data statistically, there is a significant difference, and there is nowhere near as much data in phase 3 through 5 to make a significant conclusion.

A variable I did not have high expectations for but was curious to see the result was the month end variable. Perhaps there is a difference between companies who choose a specific accounting month ending period. The findings show that there is not a significant statistical difference, although there is a greater central tendency in months such as March. There is also the turnover figure from the client data which could reflect how successful a company is. This is a very small data set, and also at best has a moderate correlation to total R&D costs. I did not expect this data set to be a useful predictor as it is generally only obtained at the point of sale, but evidently it is an effective predictor that should be useful when predicting future tax relief for the client. We should seek this data on an annual period, but also look deeper into accounting data to search for predictors.