Franchising Feasibility Valuation

Julius Kasiske

2022-09-06

1. Motivation and Tasks

In this project, an undisclosed business catering company out of Hamburg, Germany approached me regarding their plans to possible franchise their business model in order to grow their operations faster. With the experience I made in the food delivery industry with Best Bite the year prior, they asked me to analytically evaluate whether or not they should franchise their stores or not. So I had to brainstorm how I would properly depart from that task and arrive at a yes/no recommendation

1.1 General Methodology

My answer to the problem was to define both one necessary condition and a handful of sufficient conditions under which should be franchised. My necessary condition was profitability and the sufficient conditions were scalability, income regularity/safety and franchisee flexibility. Once if a possible franchise-venture would be profitable, peripheral factors like flexibility, etc. would matter, but all have to indicate favorability in order for me to give the recommendation to franchise.

1.2 Discounted Cash Flow Model

So step one was to value to what extent a franchisee's store would generate economic value. That is best approximated by looking at the stores that the company already runs itself, since on these stores, I had all the necessary data and they were run according to what the company would dictate in their franchising agreements anyways, thus they are similar in operations specifics. With the data I had, I decided to perform a discounted cash flow valuation.

Essentially, I analyzed historical income and cash flow figures in order to forecast unlevered free cash flows into the future and then discount them to their present value.

1.3 Monte Carlo Simulation

However, in DCF valuation, these forecasts are always heavily driven by assumptions. In order to make the resulting net present value more robust and contextual, I decided to perform a Monte-Carlo simulation on that DCF-model, essentially randomizing every element in the set of assumed parameters that are independent of each other in the model. The rational was if any given parameter both does not depend on other assumed parameters and its realization is not subject to any further consideration on my part, then what is to stop me from picking a value at random for each of these parameters.

While a detailed discussion of the specifics of my DCF-model is beyond the scope of this summary, I will elaborate on my work of the Monte Carlo simulation in the following.

2. Building The Simulation

2.1 Data From Excel

I built the DCF-model and its primary dash board in Excel, while turning to R for the Monte Carlo simulation. But in order to build that simulation on the model, I first had to replicate the model itself in R. That included loading the forecasted data I used for NPV calculations in Excel into R, such that I could compare the forecast in R with those out of Excel. After some wrangling and cleaning in Excel, this was the table I used:

```
##                         Name  Jahr1  Jahr2  Jahr3      Jahr8      Jahr9     Jahr10
##  1:                     Name   2022   2023   2024    2029.00    2030.00    2031.00
##  2:                  Revenue  60000  84000 117600  401953.13  502441.41  628051.76
##  3:             COGS (Mat.) -16500 -23100 -32340 -110537.11 -138171.39 -172714.23
##  4:            COGS (Pers.) -28500 -39900 -55860 -190927.73 -238659.67 -298324.58
##  5:             Gross Profit  15000  21000  29400  100488.28  125610.35  157012.94
##  6:                      D&A      0  -2000  -2000       0.00       0.00       0.00
##  7:              SGA (Raum) -10000 -15000 -15000  -25000.00  -25000.00  -25000.00
##  8:     SGA (Versicherung)  -1500  -1650  -1815   -2923.08   -3215.38   -3536.92
##  9:        SGA (Fahrzeuge) -12000 -19000 -19000  -45000.00  -45000.00  -45000.00
## 10:           SGA (Sonst.)  -1000  -1000  -1000   -1000.00   -1000.00   -1000.00
## 11:        Operating Income  -9500 -17650  -9415   26565.21   51394.97   82476.02
## 12:            Other Income      0      0      0       0.00       0.00       0.00
## 13:          Other Expenses      0      0      0       0.00       0.00       0.00
## 14:                     EBIT  -9500 -17650  -9415   26565.21   51394.97   82476.02
## 15: Unlevered Net Income  -9500 -17650  -9415   18595.64   35976.48   57733.21
## 16:                      D&A      0   2000   2000       0.00       0.00       0.00
## 17:                    CapEx  12000      0      0       0.00       0.00       0.00
## 18:                      NWC  16350  22890  32046  109532.23  136915.28  171144.10
## 19:                delta NWC  16350   6540   9156   21906.45   27383.06   34228.82
## 20:           Free Cash Flow -37850 -22190 -16571   -3310.80    8593.42   23504.39
```

Note that the forecasts for four years were purposely left out for formatting purposes. The table shows the income statement and free cash flow corrections that I forecasted in Excel. Every item (row) was forecasted individually according to a deterministic function that was developed in close collaboration with the client, after all these functions are the bedrock of what drives the NPV and should be formulated such that they model reality as closely as possible.

2.2 Testing Replicated Functionalities

Replicating the entire model in R meant I had to check whether the results I got in R were the same that I got in Excel, that way I could ensure that I simulated the correct and intended model and not some faulty version of it.

I thus coded the following testing method:

```r
test <- function(computed, actual){
  # Create Lists to compare
  intermediate <- comparison[Name == actual]
  actualList <- dtWideToVector(intermediate[, Name := NULL])
  computedList <- dtLongToVector(as.data.table(computed))
  # Create and fill boolean vector of equality tests
  equalityTest <- rep(NA, length(computed))
  for(i in 1:length(computedList)){
    errorTerm <- abs(actualList[[i]] - computedList[[i]])
    if(errorTerm <= 0.01){
      equalityTest[i] <- TRUE
    }else{
      equalityTest[i] <- FALSE
    }
  }

  # Define variables for evaluating the results
  trueCount <- 0

  # Count how many elements of the two vectors are equal
  for(i in 1:10){
    if(equalityTest[i] == TRUE){
      trueCount <- trueCount + 1
    }
  }
  elementsNotEqual <- 10-trueCount

  # Only if all ten elements are equal, allEqual is set to true
  if(trueCount == 10){
    return(TRUE)
  }else{
    print(elementsNotEqual)
    print(as.data.table(computedList))
    print(as.data.table(actualList))
    return(FALSE)
  }
}
```

This method had to cope with a variety of data structures as inputs, I thus coded conversion methods called `dtWideToVector` and `dtLongToVector` beforehand.

Note that is was written in a way such that, if the forecasts are congruent for all years, it would just return `TRUE`, while it would return all the faulty entries otherwise.

To see how it works, lets see the method in action after I replicated one line item in R. Below, I replicated the functionality for revenues and later tested it

```r
# Revenue Function
revenue <- function(Sales, COV_CAGR, CAGR){
  rev <- rep(NA, 10)
  rev[1] <- Sales
  for(i in 2:4){
    rev[i] <- round(rev[i-1]*(1+COV_CAGR), 3)
  }
  for(i in 5:10){
    rev[i] <- round(rev[i-1]*(1+CAGR), 3)
  }
  return(rev)
}
```

```r
# Test
revenueTest <- revenue(60000, 0.4, 0.25)
test(revenueTest, "Revenue")
```

```
## [1] TRUE
```

Note that since the forecasted values for revenues are the same in Excel as in R and the method returns `TRUE`, that part of the model must have been replicated correctly, and I can move on to the line items for cost types. No troubleshooting needed.

In that sequence of coding the model functionalities, testing their equality to the Excel forecasts and potentially troubleshooting, I iterated over all line items in the income statement and their free cash flow corrections, thus rebuilding the model step by step.

2.3 Random Resampling

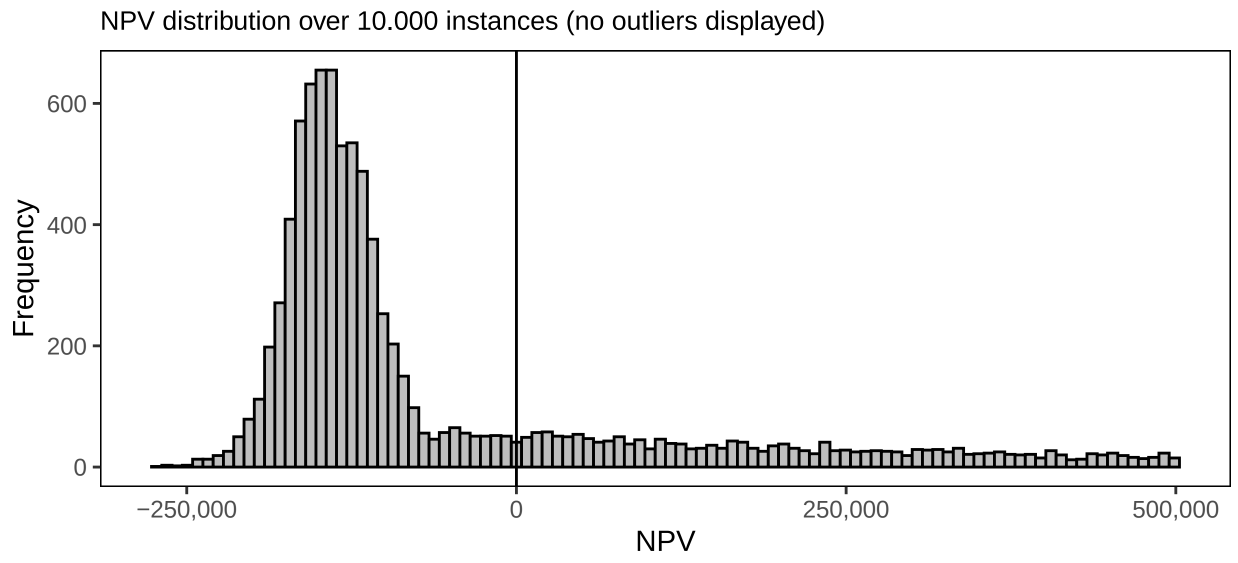
After continuing that process until a NPV could be computed in R, I had to chose the parameters to randomize, the intended distribution for each parameter and the interval in which draws should be allowed.

As mentioned above, I only randomized parameters that are both independent of the other randomized ones and would otherwise have had to be assumed. I chose uniform distributions for all parameters, as otherwise, I would just replace one assumption (the value) by more than one assumption (e.g. mean and variance of normal distribution). The intervals out of which values could be drawn was set individually for each parameter after operational considerations.

Next, I simplified the sampling by coding a function that would spare me some code later and implement the uniform draws directly. Here is the code:

```r
sampleType <- function(input){
  sample(input, 1, replace = TRUE)
}
```

Next came the bedrock of the deliverable, the simulation method. This method loops over each random parameter 10.000 times, records the resulting NPV in a vector and then plots the corresponding histogram. Here, I will spare you the code, as it contains and references large code chucks that I did not mention in detail before.



```
## [1] "Median NPV: -117277.77"
## [1] "31.48% of simulations deliver positive a NPV"
```

Note that running the method also outputs its calculations on the median NPV and the percentage of positive NPVs, giving a very compact summary of the likelihood of financial outperformance.

3. Conclusions

In this case, it was obvious that company stores themselves were run on an insufficiently viable operating model. Thus, we concluded that franchisees running that same operating model would not be able to sustain operations and the franchising model would likely not be successful for the client until it implemented an operating model that allowed for lasting profitability.

This project allowed me to merge my expertise in financial valuation, quantitative methods and, in part, the industry to arrive at analytically robust conclusions that minimized reliance on vague qualitative judgement in the evaluative process. In the end, I pointed the client in the right direction regarding their lasting value creation, sparing them countless headaches trying to implememt a franchising model that was doomed from the get-go.