CSCIE-10b

Spring 2021

**Term Project Proposal**

**Stefano Parravano**

[**parravanos@yahoo.com**](mailto:parravanos@yahoo.com)

**Abstract**

The practice of purchasing a revenue generating asset and selling it for a profit after a *brief holding period* has been the cornerstone of investment strategies for both private and institutional investors for the greater part of several decades. In real estate, this strategy is often referred to as “flipping” and it is meant to describe the process of buying, rehabbing, and selling properties for profit.

The goal of this study is to propose a model that can be used by investors to carefully evaluate potential projects. Two distinct modeling paradigms will be provided: one based on a static view of the world with fixed discrete inputs and precise deterministic computations (analytic model) and another based on a dynamic view of the world with variable inputs and stochastic calculations (simulation model).

1. **Introduction**

Real estate investing was ranked as the top investment pick for the majority of Americans, according to Gallup’s annual Economy and Person Finance survey, conducted in early April 2020[[1]](#footnote-1). While investing in the real estate asset class can be lucrative, it is certainly not devoid of risk[[2]](#footnote-2). Investing in property exposes market agents to both *systemic* and *systematic* risk; both of which need to be understood and quantified prior to the rollout of any investment strategy. This underscores the necessity to make a concerted effort to model the future of an investment proposal before committing capital resources. In recent years it has become widely accepted that computer-based algorithms can help make better and fairer investment decisions than human agents armed solely with instinct and gut feeling [[3]](#footnote-3). Pursuant to this believe, we propose a computer-based approach (algorithmic) to evaluate prospective deals in order to empower market agents with a robust decision-making process that will lead to more optimal outcomes. The framework we propose will require input from the human agent; these inputs will capture certain assumptions about the current world, and future states will be computed based on precisely defined algorithms.

1. **Methodology**

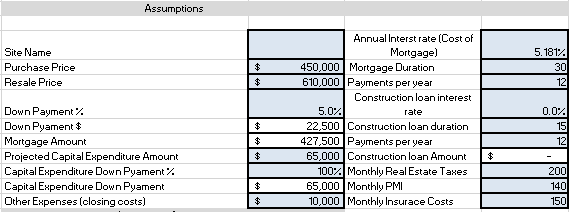
**2i. Input and Outputs**

The investor will be required to input a number of parameters that will be consumed by the modelling process. The parameters can be divided in two broad categories: those that are known *precisely* prior to the investment, and those that *are not know* and for which a future state prediction must be made by the user. The full set of these parameters will allow for a complete modelling process that will return a future state representation of the world. This future state representation will encode important information pertaining to the viability of the project and will help inform a final decision on whether capital resources should be allocated.

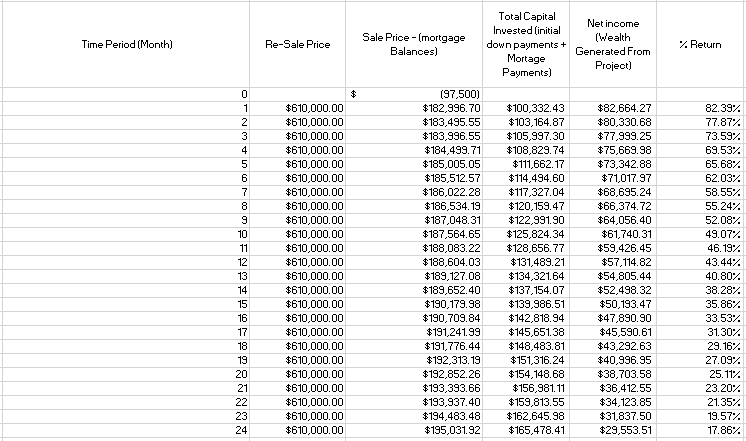
Below is a categorization of the input parameters that are required for the analytic and simulation-based models:

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Analytic Model | Simulation Model | Parameter Type |
| Purchase Price | Required | Required | Known a priori |
| Closing Costs | Required | Required | Known a priori |
| Down Payment | Required | Required | Known a priori |
| Cost of Capital (interest rate on loan) | Required | Required | Known a priori |
| Monthly Real Estate Taxes | Required | Required | Known a priori |
| Monthly Insurance Costs | Required | Required | Known a priori |
| Resale Price | Required | Required | Not known a priori |
| Resale Price Probability Distribution Parameters | Not Required | Required | Not known a priori |
| Capital Expenditure | Required | Required | Not known a priori |
| Capital Expenditure Probability Distribution Parameters | Not Required | Required | Not known a priori |
| Number of Simulations | Not Required | Required | N/A |

An interface like the one below will be made available to the user to input the various parameters:



In the case of the analytic model, the output will be a set of point estimate predictions of the net profit/ROI of the project proposal for a number of future time periods. The table below is a representation of the sort of output we will provide to the user.



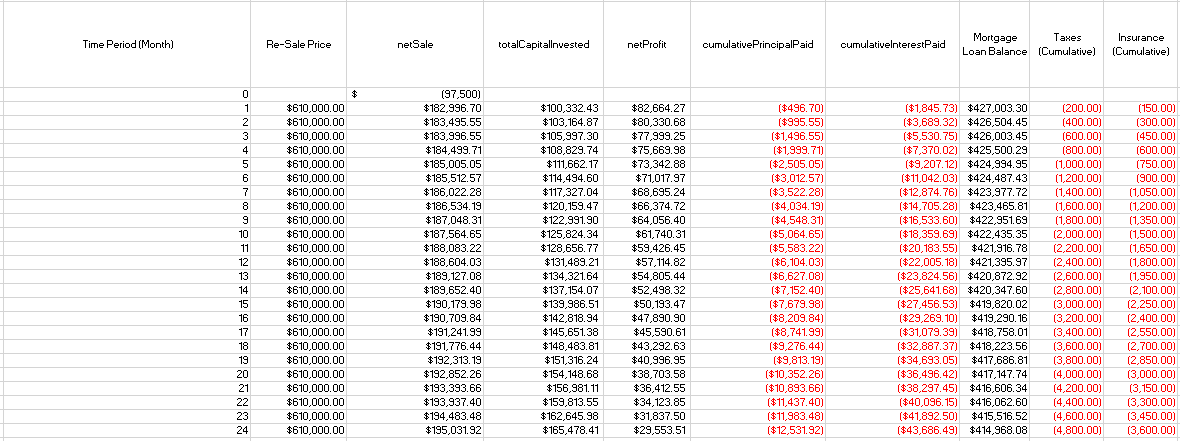
In the case of the simulation model, the same fields will be displayed but a confidence interval as opposed to a point estimates will be shown. The output gives the investor a wholistic view of profit scenarios for a wide range of future time periods. As expected, the profitability of the investment decreases over time highlighting the notion that projects completed in shorter timelines lead to higher returns. This is due to the fact that the investor must carry loan costs for each period in which the project is active. The loan interest payments must in fact be thought of as premiums the investor must pay in order to complete the project.

**2ii. Algorithms**

In order to generate the output displayed above, a number of computations must be performed. In the case of our analytic model the following operations will be completed:

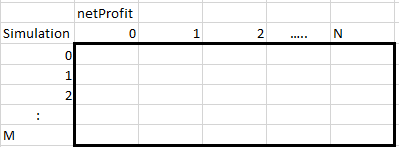
* An amortization table must be constructed
  + A fixed payment amount for each period must be computed and stored to an instance variable: **paymentAmount**
  + Principal and Interest amounts must be computed for each time period ( **+ =** for all time periods t). These amounts vary in a non-linear way across time. Results will be stored in arrays: **loanPrincipalPayment** and **interestPayment**
  + Cumulative interest and principal amounts must be computed and stored in arrays: **cumulativePrincipalPaid** and **cumulativeInterestPaid**
  + The loan balance must be computed at each time period (**loanAmount**- ) and stored to an array: **loanBalance**
* The cumulative amount of tax payments across all time periods must be computed and stored to an array:
* The cumulative amount of insurance payments across all time periods must be computed and stored to an array:
* The **netSale** values must be computed for each time period and stored to an array:
* The total amount of capital invested must be computed for each time period and stored to an array:
* **The profit at each time period can be computed as:**

Below is a visual representation off all the computations that must be performed. Each column should be thought as being stored in an array in our application:



Note: The formula for calculating the paymentAmount is given by:

The same computations will be performed for the simulation model with the added complexity that for each simulation a different value for certain key parameters will be used (drawn from a probability distribution specified by the user). Each simulation will return a netProfit array encoding the profit values for each time period across some predefined number of periods. The results from all simulation runs will be stored in an MXN 2D array where M = number of simulations and N the number of time periods we are interested in modeling (set by user). The following visual captures the essence of our simulation computations:



Once the computations are performed and stored in our data structure, a confidence interval for each time period (column level computations) will be computed.

**2iii. List of main subprograms**

The following subprograms will be needed to perform the required calculations:

* AmortizationTable
  + All required behavior to construct and compute loan amortization tables
* TotalCapitalInvested
  + All required behavior to compute the amount of capital invested at each time period
* NetProfit
  + All required behavior to combine computations from the AmortizationTable and TotalCapitalInvested routines into a final array of values meant to represent the profit generated by the investment at each time period.
* Simulation
  + All required behavior to simulate many different environments and perform the same computations in NetProfit in order to generate a probabilistic view of the possible outcome space
* FrontEndGUI
  + All functionality relating to how the interface will be constructed and displayed to the user

1. https://news.gallup.com/poll/309233/stock-investments-lose-luster-covid-sell-off.aspx [↑](#footnote-ref-1)
2. https://www.investopedia.com/articles/investing/122415/why-real-estate-risky-investment.asp#citation-1 [↑](#footnote-ref-2)
3. https://hbr.org/2020/11/do-algorithms-make-better-and-fairer-investments-than-angel-investors [↑](#footnote-ref-3)