# Multi Scenario Financial Planning via Deep Reinforcement Learning AI

Gordon Irlam
(gordoni@gordoni.com)

January 8, 2020

**Abstract**

Financial planning via deep reinforcement learning holds much promise. One implementation, AIPlanner, delivered near optimal financial results, but had a major shortcoming. It required training a separate neural network model for each financial scenario. This paper describes extending AIPlanner so that a small family of trained neural network models are capable of rapidly producing financial plans for a wide variety of financial scenarios. Additionally AIPlanner is extended to produce results over the lifecycle, both pre and post retirement, and for couples, as well as individuals. A reasonably realistic income tax model is incorporated. And finally, a more realistic stock model is used. Over the lifecycle, compared to the best discovered alternative strategy, reinforcement learning was found to effectively deliver 14% more retirement consumption.

## 1 Introduction

Reinforcement learning is a branch of machine learning. Sutton and Barto provide a good introduction to reinforcement learning. [Sutton and Barto, 2018]. Reinforcement learning has broad applicability to many problems in finance [Kolm and Ritter, 2019]. One area of application of reinforcement learning to finance is in financial planning: determining how to allocate assets and how much to consume over the lifecycle.

AIPlanner is a previously described implementation of deep reinforcement learning for financial planning [Irlam, 2018]. As previously described, for simple scenarios that have known optimal results, AIPlanner was found to deliver in excess of 98% of the best theoretical possible retirement consumption performance in every case tested.

Despite the good performance the previously described AIPlanner had a number of limitations, most significantly requiring retraining for each financial scenario considered. AIPlanner was also not able to handle pre-retirement scenarios, and ignored the effects of taxation.

## 2 Implementation

### 2.1 Multi Scenario Financial Planning

Training AIPlanner was expensive and time consuming, taking around 20 minutes for a single financial scenario. This made it difficult to deploy AIPlanner on the web, where people expect near instant results, and the longest people might be willing to wait for a page is probably around 1 minute.

| SPIAs allowed | retirement status | RRA |
|---|---|---|
| yes | any | 1.5 |
| yes | any | 3 |
| yes | any | 6 |
| no | any | 1.5 |
| no | any | 3 |
| no | pre-retirement | 6 |
| no | retired | 6 |

Table 1: Scenario sets used by AIPlanner.

Popular reinforcement learning algorithms are inherently parallelizable, and so one approach to the issue would be to parallelize training. The problem with this approach is it would require a very large computer to be constantly standing by idle just so it can handle the brief times it is actually being used.

Instead it was decided to divide the possible financial scenarios into sets of similar scenarios, and then train a single reference model for each set of scenarios. In this way when a user seeks to interact with AIPlanner, all that is necessary is to determine which set of financial scenarios their scenario belongs to, and then infer a financial plan using the model for that scenario set. Inference is fast. A financial plan can be inferred from a model in around 10 milliseconds. Although it must be said, determining how well the financial plan performs requires performing a Monte Carlo simulation, which can take anywhere from 30 to 60 seconds.

#### 2.1.1 Fine Grained Divide and Conquer

I first split the problem up based on risk aversion level, whether to allow Single Premium Immediate Annuity (SPIA) purchases, pre or post retirement status, the amount of guaranteed income relative to investments split among a number of buckets, and if SPIA purchases are allowed, the health of the individual, again split into buckets. The original AIPlanner took 2 million timesteps, or simulated years, to train. With the problem divided in this way it took 10 million timesteps to train using a variety of scenarios chosen from a given scenario set. Unfortunately, despite the best of efforts, only OK, but not stellar results were achieved.

#### 2.1.2 Coarse Grained Divide and Conquer

An expedient way to group scenarios is by the ratio of the present value of guaranteed income to portfolio investments. The relative financial plan for someone with $5k of guaranteed income per year and $100k of investments will look identical to someone with $50k of guaranteed income and $1m in investments. Combining portfolios in this way takes advantage of the iso-elasticity of the utility function. This works in the absence of taxes, but the progressive nature of income taxes means two investment portfolios will have similar strategies only if they appear equal on an after tax basis. Scenarios can be grouped like this by blinding the reinforcement learning algorithm to the absolute wealth level and dealing only with wealth levels relative to the current or initial net wealth on an after tax basis.

Good results were eventually achieved by dividing the problem up based on just the risk aversion level, whether to allow SPIA purchases, and pre or post retirement status. Training now typically takes 100 million timesteps.

Three coefficients of Relative Risk Aversion (RRA) were considered: 1.5, 3, and 6. For all SPIA purchase cases, as well as RRAs of 1.5 and 3, it was found the pre-retirement model could be used for retirement scenarios at minimal financial performance cost. This means there were 7 scenario sets as shown in Table 1.

Training, in a non-parallel fashion, now takes close to 3 days. Due to the randomness in the initial state of the machine learning process each model produces slightly different recommendations. For each scenario set I train 10 models in parallel, and at inference time take the average recommendation of the 10 models. Since inference is so fast, using an ensemble in this way delivers better results at little cost. A second motivator for training 10 models per scenario set is it enables a rough computation of the standard error associated with training. Without doing this it is impossible to tell whether a tweak to the training regime is truly beneficial, or just the result of a lucky training run. It seems plausible that rather than using an ensemble, training a single model for 1 billion years would deliver better results, but doing this would effectively require switching to a parallelized reinforcement learning algorithm. This has its own set of positives and negatives.

The total cost of training the 70 models for typically 100 million time steps on Amazon EC2 on-demand instances in 2019 is around $400 dollars, although cheaper spot instances can be used for as little as one quarter the price.

## 2.2 Financial Model Improvements

### 2.2.1 Planning Over the Full Lifecycle

Previously AIPlanner was restricted to planning only in retirement. AIPlanner is now capable of planning pre-retirement. Beyond simply simulating the pre-retirement years, this involved adding liability modeling to AIPlanner for handling items such as mortgages, student loans, auto loans, children/dependents, anticipated college expenses, and home down payments, as well as new assets such as wages.

In pre-retirement planning, a decision has to be made. Do you only determine how much to consume post retirement, or do you determine the ideal consumption over the entire lifecycle including pre-retirement? By default, AIPlanner uses a fixed consumption amount pre-retirement, and only plans consumption in retirement. This is what most people expect from a tool intended to help with planning for retirement. However, by setting the "retirement age" to the starting age, and having wage income that ends at the true retirement age, it is possible to derive consumption recommendations over the full lifecycle.[1] Asset allocation recommendations are made over the full life cycle in either case.

### 2.2.2 Couples

Accurately handling financial planning for couples is important, but it is something AIPlanner has struggled with in the past. The current strategy is to only train models for a single individual. Subsequently a couple is treated like a single individual but with a retirement expectancy equal to the number of years for which both members of the couple are expected to be alive in retirement plus the number of years one member is expected to be alive divided by 1.6. The rationale for this is two can live together as cheaply as 1.6 individuals alone. Utility of consumption for the couple is given by the utility of consumption divided by 1.6, but carries twice the weight of consumption by a single individual.

The lack of suitable benchmarks makes it difficult to assess how well this approach to handling couples works. One area where it is likely to be weak is in making SPIA annuitization decisions. A couple gets part of the benefits of annuitization without having to annuitize, simply by virtue of the fact that when one member dies, the other receives the portfolio wealth that had been notionally set aside to cover the needs of the first individual. Consequently AIPlanner might currently be over aggressive in recommending SPIAs to couples. Experiments showed that ignoring annuitization recommendations until only one member is alive produced worse overall well being for a couple than following the recommendations. So it seems likely that this concern is not a major one.

[1] AIPlanner hasn't been trained with retirement ages less than 50, so the performance of younger retirement ages is uncertain.

### 2.2.3 Taxes

AIPlanner now includes a fairly detailed tax model. Both income and wealth are divided into tax free, tax deferred, and taxable amounts. Individual allocations to each asset class are maintained for taxable assets. For simplicity average cost is used as the cost basis method for changes in share holdings.

The tax model is currently based on the U.S. income tax code for 2020. Many simplifications are obviously made, but the level of detail is still substantial. A progressive tax rate structure is used with the standard deduction, special taxation rules applied to Social Security payments, capital gains tax rates, capital loss carry forwards, and the net investment income tax. 401(k) and IRA contributions are allowed and the IRA catch-up amount is available for older workers.

State and local taxes on the other hand are greatly simplified. An average state and local tax rate equivalent to 4% of income, along with an average property tax rate equivalent to 2% of income are applied. Although, in hindsight, 3% probably would have been a more accurate assessment of property tax burden. These figures reflect national averages. Improving much beyond this would require diving into the details of how taxes are assessed in every state. This would be a formidable task.

## 2.3 Asset Class Returns

### 2.3.1 Domestic Stocks

The overall U.S. stock market returns are based on the historical returns reported by the 2019 Credit Suisse yearbook [Dimson et al., 2019]: 6.5% real arithmetic mean and 17.4% standard deviation (real geometric mean 5.0%). However, unlike stochastic dynamic programming, we are able to go beyond the headline statistics and embed both a mean reversion model and a volatility model into these returns.

Shiller [Shiller, 2015] argues that stocks at times exceed or fall below their rational valuations. I find these arguments persuasive. Consequently for every 10% stocks are over-valued I apply a 1% decline in expected return. This is based on eyeballing the results of Shiller. Doing this raises the important question in simulating stock returns of what is the fair value. 30% of the volatility in stock prices is attributed to changes in the fair value, and 70% to irrational exuberance or pessimism. This value was chosen to produce a reasonable looking price to fair value distribution, with stocks priced at 50 - 150% of fair value for the vast majority of the time. This is shown in Figure 1. The estimates used here are certainly rough. It is simply a question of, if by using them, we improve the accuracy of the model over assuming standard fixed returns. I believe this to be the case.

Stock volatility uses a GJR-GARCH model [Wikipedia contributors, 2019a] in which volatility varies over time. Applying a daily GJR-GARCH model is computationally too expensive, so I calibrated a monthly GJR-GARCH model to the historical returns over the period 1970 to 2018. The residuals used to provide the stochastic returns were the residuals from this calibration exercise, meaning that returns are not log normally distributed but instead display fat tails; i.e. skew and kurtosis.

An example synthetic stock market price index and fair value index is shown in Figure 2.

When evaluating performance stocks are initially assumed to be fairly valued with a volatility equal to the average volatility.

AIPlanner is able to observe the stock price to fair value and the trailing volatility. However to be fair, a 20% standard deviation noise factor is added to the observed stock price to reflect the difficulty in assessing whether stocks are truly fairly valued.

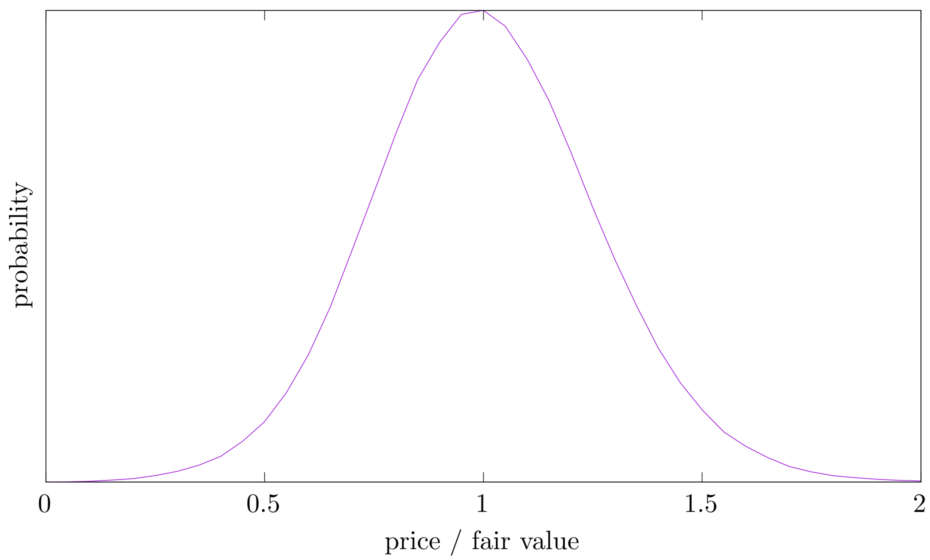


Figure 1: Distribution of ratio of stock price to fair value.

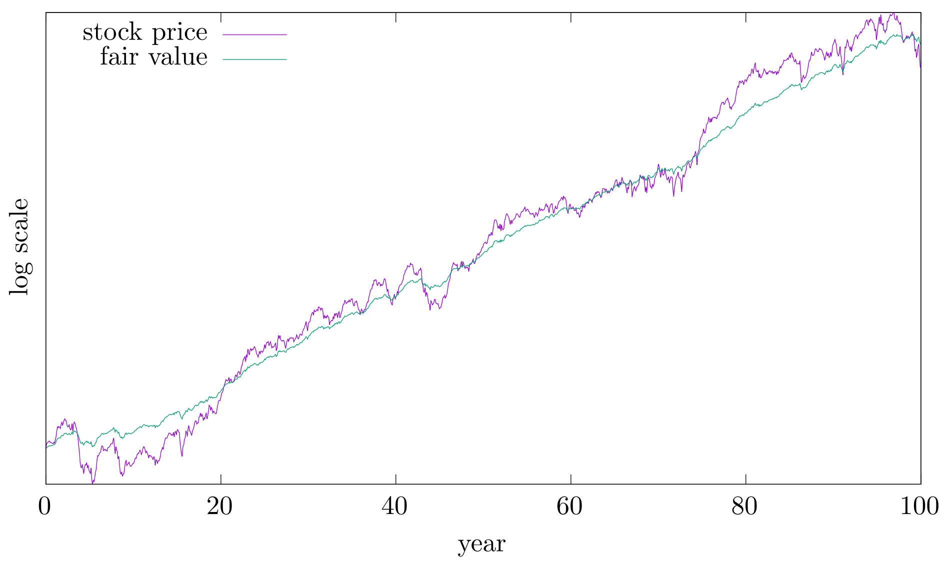


Figure 2: Example synthetic stock price index and fair value index.

### 2.3.2 Bonds

As previously described the bond model used is a Hull-White model [Wikipedia contributors, 2019c], meaning that there is a yield curve and the yield curve varies over time in a non-linear fashion depending on changes to the short interest rate. The dynamics of the Hull-White yield curve were calibrated to the those of the U.S. Treasury yield curves over the period 2005 - 2017, with the yield curve on 2018-12-31 taken as a typical yield curve. The initial short real rate was 1.0%, and the initial inflation rates was 1.6%. The Hull-White model is used to provide both the real interest rate and the inflation rate. Nominal rates are constructed from these two assuming the break even inflation rate is equal to the inflation rate.

The Hull-White model means bonds are less risky than they might at first appear. Returns exhibit a degree of mean reversion. When rates go up, producing poor returns, it is more likely that they will fall in the future, producing good returns, and vice versa.

A deficiency of the Hull-White model is that the short term rates tend to be more volatile than observed in practice. A Heath-Jarrow-Morton bond model [Wikipedia contributors, 2019b] could probably fix this, but at considerable computational cost, and so has not been implemented.

No attempt is made to prevent the nominal interest rate falling below zero. This has been observed to happen in the financial model sometimes.

AIPlanner is able to observe the real interest rate.

### 2.3.3 International Stocks

While domestic stocks and bonds are two fundamentally different asset classes, domestic stocks and international stocks have correlated statistical models that behave very similarly. As shown later when looking at using valuations to attempt to time the market between stocks and bonds, valuation based market timing can be challenging. As a result I am pessimistic of being able to time the international market, switching between domestic and international stocks based on valuations. There is also the possible loss of foreign tax credits. The vast majority of international funds in the U.S. are unhedged. This implies an increased currency induced volatility of international stocks. For these reasons no attempt has been made to model international stocks.

It is fine to invest in international stocks. Sometimes you will win and sometimes you will lose, but probably don't expect an outcome that is significantly different on average from investing in domestic stocks alone.

## 2.4 Asset Class Standard Errors

With little more than 100 years of high quality returns data the uncertainty in the mean return is substantial. Consequently, based on the 2019 Credit Suisse yearbook I apply a standard error of 1.6% to stock returns. This means a majority of the training episodes will get an expected mean return value that is within 1.6% of the measured value from the Credit Suisse yearbook. To this expected mean return the volatility model is applied.

For real bonds a 1.0% standard error is applied to the average long run short rate. Meaning that a majority of episodes will have an average short rate that is within 1.0% of the average short rate of the calibration period.

For inflation a standard error of 1.6% is used.

## 2.5 Reinforcement Learning

### 2.5.1 Reinforcement Learning Model Observations

AIPlanner judiciously chooses what properties of the financial model the reinforcement learning framework gets to observe. Too few choices and reinforcement learning will be unable to optimize effectively. Too many and the reinforcement learning algorithm will identify spurious correlations amongst disparate data points.

The key observations made of the financial model are:

- Age related observations:
  - The number of years remaining pre-retirement.
  - An estimate of the expected number of years remaining in retirement.
- Annuitization related observations:
  - The 80th percentile remaining lifespan.
  - The life expectancy assumed by the insurance company associated with a SPIA purchased at the current age.
  - Whether it is the final year in which SPIA purchases are allowed.
- Wealth related observations:
  - The fraction of total wealth comprised of investment portfolio wealth.
  - The fraction of total wealth comprised of pre-retirement wealth.

  Total wealth is comprised of investment portfolio wealth plus the expected net present value of guaranteed income computed using a discount rate of zero and both adjusted for expected future taxes.
- Market related observations:
  - A noisy estimate of the current stock price to fair value.
  - The trailing stock market volatility.
  - The real interest rate.

  It is hoped that by observing these parameters AIPlanner can perform tactical asset allocation and consumption.
- Consumption and reward observations:
  - A rough normalized estimate of expected consumption level over the remainder of the episode, computed as total wealth divided by retirement years remaining.
  - An estimate of the expected utility remaining for this episode; i.e. the "reward to go".

  It is unusual that we have access to this information in a reinforcement learning framework, but using it has been found to deliver better results.
- Other observations:
  - Whether a 401(k) is available.

#### 2.5.2 Reinforcement Learning Model Actions

The neural network model has to predict four actions to take in each time period:

- Consumption: I start with a rough consumption estimate of net wealth divided by retirement years remaining, and require the model pick a value between 0.1 and 1.9 times this amount. The mid-point of 1.0 thus serves as the initial training default value.
- SPIA annuitization amount: I have the model try and learn the ideal fraction of net wealth that should be held in guaranteed income. I then compute the difference between this fraction and the current fraction of net wealth held in guaranteed income. If this difference is 10% or more, and SPIA purchases are allowed, I purchase that amount in SPIAs.
- Asset allocation: I had difficulty trying to accurately predict the investment portfolio stock allocation, $\pi_p$, so I use as a starting point the knowledge that the investment portfolio stock allocation curves upwards as the ratio of guaranteed income wealth, $W_g$, to investment portfolio wealth, $W_p$, increases. In other words, I have the reinforcement learning model attempt to predict the $x$ in,

  $$\pi_p = x + c\frac{W_g}{W_p}$$

  where $c$ might also be variable, or a constant. Trying to learn both $x$ and $c$ was difficult, so I am currently using a fixed value of 0.5 for $c$. This equation has the desirable properties that $\pi_p \to \infty$ as $W_g \to \infty$, and $\pi_p \to x$ as $W_p \to \infty$.
- Bond duration: The model simply determines a duration between 1 and 30 years.

#### 2.5.3 Reinforcement Learning Framework

For the reinforcement learning framework I switched from using the OpenAI Baselines PPO algorithm [Schulman et al., 2017] to the Ray RLlib [Liang et al., 2018] implementation of PPO. Ray is able to take advantage of cheaper spot compute instances. In addition to PPO, Ray also includes an implementation of the newer IMPALA reinforcement learning algorithm [Espeholt et al., 2018]. Some quick experiments showed PPO outperforming IMPALA on financial planning problems, and so PPO continues to be the algorithm of choice. The removal of running mean observation normalization as occurs in the OpenAI Baselines implementation also probably helped boost results.

For 100 million time steps the optimizer step size is set to 5e-6, and the batch size is 200,000. These values were chosen on account of the high stochasticity of the financial problem.

I am now training a 256 x 256 fully connected net, as opposed to a 64 x 64 fully connected net previously. This is computationally quite a bit more expensive, but necessary to get good results with the wide range of scenarios within a scenario set.

## 3 Results

Results were produced by first training the neural network models using reinforcement learning, a process which can take close to 3 days, and then evaluating their performance using Monte Carlo simulation, a process which takes 10 - 15 minutes. Evaluations were performed for 2 million timesteps which provides for a high degree of accuracy.

All results reported here are for a single female having a specified age in 2020. AIPlanner is trained for a variety of ages and health possibilities, and age and health are key factors in determining the suitability of SPIAs. People interested in financial planning tend to be in better health than their cohort. This is reflected when evaluating by increasing the life expectancy of the scenarios by 3 years over that specified by the Social Security cohort actuarial tables.

Both during training and evaluation the full probabilistic range of life expectancy possibilities are considered. There is no fixed lifespan or Monte Carlo lifespan rollouts.

A variety of different scenarios are considered during training. During evaluation retirement is assumed to occur at age 67, and Social Security payments of $20,000 per year are assumed from that date. Prior to this, unless otherwise specified, a constant labor income of $50,000 is assumed and a constant $30,000 of consumption occurs. During training the individual may or may not have a 401(k) available. If they do not they are assumed to have an IRA available. During evaluation a 401(k) was assumed to be present.

During evaluation it is necessary to come up with an initial investment portfolio. I split the initial investment amount up as 25% tax free, 50% tax deferred, and 25% taxable, with the later split 60/40 between stocks and bonds with cost basis fractions of 100%.

Risk aversion is a personal decision. Reflecting this I frequently consider three different RRA values: 1.5, 3, and 6.

Performance is reported as Certainty Equivalents (CEs), which boil down the variable sea of consumption numbers to a single overall number.

The client has an unlimited psychological ability to handle changes in nominal investment portfolio value. The client's goal is simply to maximize expected utility of consumption over their lifetime.

## 3.1 Validation Results

A version of AIPlanner was trained with log-normal Independent Identically Distributed (IID) stock and bond returns, and no standard error. Returns were chosen to largely match the returns reported in the 2019 Credit Suisse year book: 6.5% arithmetic mean return and 17.4% volatility for stocks, and a lower 1.0% mean return reflective of current times, and 11.0% volatility for bonds. SPIAs were not allowed. For such a simple scenario it is possible to calculate the exact optimal results using stochastic dynamic programming.

AIPlanner was trained using the full range of taxable asset possibilities. When evaluating performance the individual was imbued with different specific initial amounts of tax free investment assets; i.e. Roth assets. Additionally when evaluating, labor income and Social Security were treated as tax free.[2]

Two different sets of result were generated. One for a working individual age 50, and one for an individual just retired at age 67. The results are shown in Table 2. The key column to look at is the column labeled relative performance.

The relative performance column shows AIPlanner exceeding the best theoretical possible results at age 50, RRA 1.5, for four different initial portfolio sizes. Obviously this is not possible. I chose to investigate the worst offender: initial tax free assets of $500,000, relative performance 100.4%. For this evaluation the standard error of CE measurement was estimated to be 0.3%, so the result lies well within two standard deviations of 100.0%, suggesting that it could just be a measurement error. This was born out by altering the random number seed for the evaluations and re-running the evaluations. This resulted in the performance dropping to 99.4%. The same sequence of random returns is generated for all scenarios with a given initial

[2] I accidentally carried over into later result evaluations this tax free attribute for Social Security during evaluation. This doesn't invalidate these later results as they involved comparisons between other similar evaluations. During training Social Security is always taxable.

| age | RRA | initial tax free assets | Optimal CE | initial stocks | initial consume | AIPlanner CE | initial stocks | initial consume | relative performance |
|---|---|---|---|---|---|---|---|---|---|
| 50 | 1.5 | 200,000 | 78,718 | 100% | - | 78,955 | 100% | - | 100.3% |
| 50 | 1.5 | 500,000 | 113,979 | 100% | - | 114,385 | 100% | - | 100.4% |
| 50 | 1.5 | 1,000,000 | 170,639 | 100% | - | 171,153 | 100% | - | 100.3% |
| 50 | 1.5 | 2,000,000 | 281,366 | 100% | - | 281,541 | 100% | - | 100.1% |
| 50 | 3 | 200,000 | 64,466 | 100% | - | 64,394 | 100% | - | 99.9% |
| 50 | 3 | 500,000 | 87,012 | 100% | - | 86,628 | 100% | - | 99.6% |
| 50 | 3 | 1,000,000 | 121,556 | 100% | - | 120,267 | 100% | - | 98.9% |
| 50 | 3 | 2,000,000 | 186,730 | 97% | - | 182,337 | 100% | - | 97.6% |
| 50 | 6 | 200,000 | 51,269 | 100% | - | 50,661 | 100% | - | 98.8% |
| 50 | 6 | 500,000 | 64,495 | 96% | - | 62,930 | 100% | - | 97.6% |
| 50 | 6 | 1,000,000 | 84,268 | 79% | - | 80,573 | 100% | - | 95.6% |
| 50 | 6 | 2,000,000 | 120,481 | 70% | - | 112,085 | 90% | - | 93.0% |
| 67 | 1.5 | 200,000 | 32,728 | 100% | 26,932 | 32,615 | 100% | 28,560 | 99.7% |
| 67 | 1.5 | 500,000 | 49,867 | 100% | 42,487 | 49,802 | 100% | 42,197 | 99.9% |
| 67 | 1.5 | 1,000,000 | 77,350 | 100% | 67,200 | 77,346 | 100% | 64,856 | 100.0% |
| 67 | 1.5 | 2,000,000 | 131,058 | 100% | 115,264 | 131,096 | 100% | 109,860 | 100.0% |
| 67 | 1.5 | 5,000,000 | 289,779 | 100% | 256,892 | 288,922 | 99% | 244,500 | 99.7% |
| 67 | 3 | 200,000 | 31,508 | 100% | 29,892 | 31,503 | 100% | 29,643 | 100.0% |
| 67 | 3 | 500,000 | 46,023 | 100% | 45,266 | 45,976 | 100% | 42,615 | 99.9% |
| 67 | 3 | 1,000,000 | 68,246 | 100% | 68,663 | 68,039 | 100% | 63,347 | 99.7% |
| 67 | 3 | 2,000,000 | 110,318 | 92% | 112,562 | 109,305 | 99% | 103,861 | 99.1% |
| 67 | 3 | 5,000,000 | 232,168 | 85% | 239,241 | 224,735 | 94% | 224,275 | 96.8% |
| 67 | 6 | 200,000 | 29,987 | 100% | 30,262 | 29,880 | 100% | 28,758 | 99.6% |
| 67 | 6 | 500,000 | 41,316 | 83% | 42,853 | 40,968 | 100% | 40,286 | 99.2% |
| 67 | 6 | 1,000,000 | 58,040 | 72% | 61,150 | 57,286 | 90% | 57,811 | 98.7% |
| 67 | 6 | 2,000,000 | 88,794 | 65% | 94,682 | 87,060 | 83% | 90,184 | 98.0% |
| 67 | 6 | 5,000,000 | 174,720 | 60% | 188,503 | 166,307 | 80% | 183,816 | 95.2% |

Table 2: Performance of the theoretically optimal results and AIPlanner. This table shows the certainty equivalents, initial stock asset allocations, and initial consumption amounts for a number of log-normal IID scenarios.

age. This explains why all four scenarios were impacted.

AIPlanner was consistently found to over estimate the initial stocks asset allocation by around 20%. For low coefficients of relative risk aversion, and small investment portfolio sizes this doesn't have a significant effect on the certainty equivalent. Small inaccuracies in asset allocation will have very little impact, but 20% is starting to become large. I will explore the impact of this bias and how it can be corrected in detail later this document.

AIPlanner performs very well overall, but does worse at higher coefficients of relative risk aversion, for large portfolio sizes, and for pre-retirement. This is to be expected. A high coefficient of relative risk aversion makes the scenario less forgiving of errors in asset allocation or consumption. A large portfolio size relative to the fixed Social Security guaranteed income amount means there is relatively less of a cushion to fall back on should the investment portfolio start to become depleted. Modeling pre-retirement scenarios is a fundamentally harder problem than modeling retirement scenarios alone. The pre-retirement model includes what to do during retirement as a sub-component. Since a pre-retirement individual would switch to using the retirement model upon retirement, pre-retirement results understate the results obtainable using AIPlanner.

In summary for the simplified stock and bond model AIPlanner usually delivers results that are within 1 - 2% of the best theoretically possible results, but can do worse at high levels of risk aversion if the ratio of the investment portfolio to guaranteed income is also very large.

This is highly encouraging. AIPlanner knows nothing of identically distributed log-normal returns. Stock and bond returns are just sources of randomness to the reinforcement learning algorithm. It suggests that when we switch to richer stock and bond models we can hope to obtain similarly good performance. It is a hope rather than a guarantee because AIPlanner may fail to exploit opportunities that these richer models contain. In particular, taking advantage of mean reversion and variable volatility.

## 3.2 Real Bonds

Real bonds, or TIPS, are bonds that return a guaranteed inflation adjusted amount. It is often said that stocks are for return, bonds are for stability. Nominal bonds carry inflation uncertainty. Real bonds don't. The additional cost of real bonds is very small, or in my model zero. This means real bonds should be preferred.

The real bond yield curve is based on the Treasury yield curve. The nominal bond yield curve is based on adding inflation to the real yield curve. In the real world it is also necessary to add the inflation risk premium less the nominal liquidity premium, but these two together are small. For instance, the 2019-12-31 10 year break even inflation rate is 1.7%, while the 2019Q1 Survey of Professional Forecasters 10 year inflation expectation is 2.2%. Thus if the Survey of Professional Forecasters is to be believed the inflation risk premium less the liquidity premium is -0.5%. Real bonds are cheaper than nominal bonds! A market based comparison would thus be between my nominal bonds minus 0.5% and real bonds.

Corporate bonds offer the promise of higher yields than Treasury bonds, but with the risk of defaults and downgrades. The most common bond rating is A. From 1997 to 2018 the average value of the BofA Merrill option adjusted spread for A rated bonds over Treasuries is 1.3%. This includes very high spreads during the financial crisis. I estimate the cost of defaults and downgrades for A rated bonds as 0.5%. This suggests a fair comparison for corporate bonds might be between my nominal bonds plus 0.8% and real bonds, or my nominal bonds plus 0.3% and real bonds if real bonds are indeed cheaper than nominal bonds.

Table 3 shows the results of training and evaluating the full rich stock and bond models using nominal bonds, nominal bonds plus 1%, and real bonds for a RRA of 6.

For small portfolio sizes the differences are small. This is because these portfolios are likely to be close to 100% stock most of the time. The improvement from using real bonds in place of nominal bonds shows up

| age | RRA | initial assets | nominal bonds CE | nominal bonds plus 1% CE | improvement over nominal | real bonds CE | improvement over nominal |
|---|---|---|---|---|---|---|---|
| 67 | 6 | 200,000 | 28,880 | 28,911 | 0.1% | 29,730 | 2.9% |
| 67 | 6 | 500,000 | 38,942 | 39,447 | 1.3% | 41,367 | 6.2% |
| 67 | 6 | 1,000,000 | 54,306 | 55,938 | 3.0% | 59,292 | 9.2% |
| 67 | 6 | 2,000,000 | 81,172 | 84,816 | 4.5% | 91,274 | 12.4% |
| 67 | 6 | 5,000,000 | 150,754 | 160,167 | 6.2% | 177,905 | 18.0% |

Table 3: Improvement from the use of nominal bonds plus 1% and real bonds over nominal bonds for a number of rich stock and bond model scenarios.

the most for larger portfolios, and were observed to be as large as 18%, albeit for a $5m portfolio.

Turning to corporate bonds. The table shows that even if one buys into the theory that corporate bonds offer higher returns, they would still be substantially inferior to real bonds.

AIPlanner selects the bond duration. For real bonds it typically selects a duration of around 20 years. This makes sense as, other things being equal, it is desirable to match bond duration to the duration of liabilities to be funded, and longer duration bonds usually have higher yields.

It wasn't measured, but for lower levels of risk aversion, the advantages of real bonds are likely to be less.

As a result of the advantages of real bonds, unless otherwise mentioned, I will use real bonds exclusively to assess the performance of AIPlanner.

## 3.3 SPIAs

An unexpectedly long life is one of the best possibilities to be wished for, but if it means your investment portfolio has become depleted it can also become one of the worst. Single premium immediate annuities and deferred income annuities provide a form of longevity insurance.

In AIPlanner nominal SPIAs are purchased with a fixed 2% annual inflation adjustment. SPIAs aren't allowed to be purchased pre-retirement, and can be purchased gradually with a minimum purchase amount of 10% of total wealth, including the net present value of guaranteed income. SPIAs can only be purchased through age 85. Nominal SPIAs are priced at a Money's Worth Ratio relative to nominal bonds of 100%, reflective of the competitive U.S. marketplace.[3]

Table 4 presents the performance of AIPlanner both with and without SPIAs being available for an individual in good health. Improvements from using SPIAs are substantial, especially for larger portfolios and higher levels of risk aversion.

## 3.4 Impact of Stock and Bond Models

Mean reversion of stock and bond returns as well as observation of the current stock price, volatility, and real interest rate should all help AIPlanner outperform the results from a more naive IID stock and bond model.

Table 5 presents the results of evaluation with stock and bond returns sampled at random from the same

[3] At a MWR of 100% insurance companies can still make money by earning the difference between Treasury nominal bond yields and corporate bond yields, but in doing so take on some default risk.

| age | RRA | initial assets | no SPIAs CE | nominal SPIAs CE | improvement |
|---|---|---|---|---|---|
| 50 | 1.5 | 200,000 | 65,394 | 68,917 | 5.4% |
| 50 | 1.5 | 500,000 | 93,713 | 100,662 | 7.4% |
| 50 | 1.5 | 1,000,000 | 137,852 | 149,806 | 8.7% |
| 50 | 1.5 | 2,000,000 | 221,002 | 243,329 | 10.1% |
| 50 | 3 | 200,000 | 58,542 | 62,443 | 6.7% |
| 50 | 3 | 500,000 | 80,769 | 87,599 | 8.5% |
| 50 | 3 | 1,000,000 | 114,661 | 127,688 | 11.4% |
| 50 | 3 | 2,000,000 | 176,564 | 202,133 | 14.5% |
| 50 | 6 | 200,000 | 48,249 | 53,251 | 10.4% |
| 50 | 6 | 500,000 | 62,505 | 72,361 | 15.8% |
| 50 | 6 | 1,000,000 | 84,603 | 101,295 | 19.7% |
| 50 | 6 | 2,000,000 | 126,077 | 159,083 | 26.2% |
| 67 | 1.5 | 200,000 | 31,797 | 32,336 | 1.7% |
| 67 | 1.5 | 500,000 | 47,195 | 48,760 | 3.3% |
| 67 | 1.5 | 1,000,000 | 70,977 | 74,757 | 5.3% |
| 67 | 1.5 | 2,000,000 | 115,199 | 125,094 | 8.6% |
| 67 | 1.5 | 5,000,000 | 236,859 | 265,831 | 12.2% |
| 67 | 3 | 200,000 | 30,978 | 31,656 | 2.2% |
| 67 | 3 | 500,000 | 44,741 | 46,755 | 4.5% |
| 67 | 3 | 1,000,000 | 65,693 | 70,920 | 8.0% |
| 67 | 3 | 2,000,000 | 104,052 | 117,127 | 12.6% |
| 67 | 3 | 5,000,000 | 207,756 | 247,384 | 19.1% |
| 67 | 6 | 200,000 | 29,730 | 30,437 | 2.4% |
| 67 | 6 | 500,000 | 41,367 | 43,931 | 6.2% |
| 67 | 6 | 1,000,000 | 59,292 | 65,813 | 11.0% |
| 67 | 6 | 2,000,000 | 91,274 | 107,517 | 17.8% |
| 67 | 6 | 5,000,000 | 177,905 | 223,525 | 25.6% |

Table 4: Certainty equivalent performance improvement in the presence of nominal SPIAs for a number of rich stock and bond model scenarios for an individual in good health.

| SPIAs | age | RRA | initial assets | mean reverting stocks and bonds CE | IID stocks and mean reverting bonds CE | mean reverting stocks and IID bonds CE |
|---|---|---|---|---|---|---|
| none | 67 | 6 | 200,000 | 29,730 | 29,787 | 28,890 |
| none | 67 | 6 | 500,000 | 41,367 | 40,635 | 39,076 |
| none | 67 | 6 | 1,000,000 | 59,292 | 57,183 | 54,450 |
| none | 67 | 6 | 2,000,000 | 91,274 | 87,483 | 81,386 |
| none | 67 | 6 | 5,000,000 | 177,905 | 168,883 | 151,986 |

Table 5: Certainty equivalent performance of various stock and bond model scenarios.

distribution as the more sophisticated stock and bond models. This isn't a completely fair comparison as I am using the asset allocation and consumption recommendations for the sophisticated model with returns sampled at random, rather than training a new model for this case. I did this because training a new model is expensive. Results here are thus indicative rather than precise.

The difference in performance from the different stock and bond model assumptions isn't massive, but it does make a difference, with the more sophisticated stock and bond models yielding better results. The bond model used has a bigger impact than the stock model used.

## 3.5 Bias Assessment and Correction

As mentioned previously the initial asset allocation results of AIPlanner are consistently biased in favor of stocks, by about 20% for log-normal IID returns, and for whatever reason, by a lower amount for the richer stock and bond models. This bias is not due to the tax model. Bias continues to exist in a model trained without any taxes. Instead it is suspected that it could be a result of the credit assignment problem in reinforcement learning. It isn't always possible to know whether poor results experienced now are the result of over aggressive asset allocation occurring many time periods earlier.

It is possible to correct for the bias by subtracting a fixed amount from all of the asset allocation recommendations of each model. The improvement from doing so for the rich stock and bond models both in the absence and presence of nominal SPIAs are shown in Tables 6 and 7.

For most models the improvement from correcting for bias are insignificant. It is only for the pre-retirement models with an RRA of 6 that the correction has a significant payoff. This includes the results for the SPIAs, age 67, RRA 6 scenario since here the pre-retirement and retired scenarios share the model in the presence of SPIAs. Note that the no SPIA scenarios where the improvements are the greatest are exactly the scenarios where AIPlanner showed the weakest performance in the log-normal IID case: age 50, RRA 6.

For the two models in which bias correction is identified as having a significant payoff I correct for the bias in the deployed version of AIPlanner. It isn't necessary to manually reduce the predicted asset allocations shown on the AIPlanner website.

## 3.6 Comparison to Common Advice

In this section I explore the results of AIPlanner over the lifecycle compared to common financial planning advice.

The starting point is a 30 year old female investor with no assets other than a future promise of $20k per year of Social Security in retirement at age 67. They earn $80k per year before tax, and consume $50k. A

| SPIAs | age | RRA | initial assets | CE | bias applied | corrected CE | improvement |
|---|---|---|---|---|---|---|---|
| none | 50 | 1.5 | 200,000 | 65,394 | 5% | 65,321 | -0.1% |
| none | 50 | 1.5 | 500,000 | 93,713 | 5% | 93,601 | -0.1% |
| none | 50 | 1.5 | 1,000,000 | 137,852 | 5% | 137,853 | 0.0% |
| none | 50 | 1.5 | 2,000,000 | 221,002 | 5% | 221,819 | 0.4% |
| none | 50 | 3 | 200,000 | 58,542 | 0% | - | - |
| none | 50 | 3 | 500,000 | 80,769 | 0% | - | - |
| none | 50 | 3 | 1,000,000 | 114,661 | 0% | - | - |
| none | 50 | 3 | 2,000,000 | 176,564 | 0% | - | - |
| none | 50 | 6 | 200,000 | 48,249 | -15% | 49,420 | 2.4% |
| none | 50 | 6 | 500,000 | 62,505 | -15% | 64,819 | 3.7% |
| none | 50 | 6 | 1,000,000 | 84,603 | -15% | 88,033 | 4.1% |
| none | 50 | 6 | 2,000,000 | 126,077 | -15% | 128,356 | 1.8% |
| none | 67 | 1.5 | 200,000 | 31,797 | 5% | 31,796 | -0.0% |
| none | 67 | 1.5 | 500,000 | 47,195 | 5% | 47,170 | -0.1% |
| none | 67 | 1.5 | 1,000,000 | 70,977 | 5% | 70,929 | -0.1% |
| none | 67 | 1.5 | 2,000,000 | 115,199 | 5% | 115,355 | 0.1% |
| none | 67 | 1.5 | 5,000,000 | 236,859 | 5% | 238,228 | 0.6% |
| none | 67 | 3 | 200,000 | 30,978 | 0% | - | - |
| none | 67 | 3 | 500,000 | 44,741 | 0% | - | - |
| none | 67 | 3 | 1,000,000 | 65,693 | 0% | - | - |
| none | 67 | 3 | 2,000,000 | 104,052 | 0% | - | - |
| none | 67 | 3 | 5,000,000 | 207,756 | 0% | - | - |
| none | 67 | 6 | 200,000 | 29,730 | -5% | 29,749 | 0.1% |
| none | 67 | 6 | 500,000 | 41,367 | -5% | 41,562 | 0.5% |
| none | 67 | 6 | 1,000,000 | 59,292 | -5% | 59,435 | 0.2% |
| none | 67 | 6 | 2,000,000 | 91,274 | -5% | 91,323 | 0.1% |
| none | 67 | 6 | 5,000,000 | 177,905 | -5% | 178,109 | 0.1% |

Table 6: Certainty equivalent performance improvement after correcting for bias for a number of rich stock and bond model scenarios without SPIAs.

| SPIAs | age | RRA | initial assets | CE | bias applied | corrected CE | improvement |
|---|---|---|---|---|---|---|---|
| nominal | 50 | 1.5 | 200,000 | 68,917 | 0% | - | - |
| nominal | 50 | 1.5 | 500,000 | 100,662 | 0% | - | - |
| nominal | 50 | 1.5 | 1,000,000 | 149,806 | 0% | - | - |
| nominal | 50 | 1.5 | 2,000,000 | 243,329 | 0% | - | - |
| nominal | 50 | 3 | 200,000 | 62,443 | -5% | 62,663 | 0.4% |
| nominal | 50 | 3 | 500,000 | 87,599 | -5% | 87,913 | 0.4% |
| nominal | 50 | 3 | 1,000,000 | 127,688 | -5% | 128,285 | 0.5% |
| nominal | 50 | 3 | 2,000,000 | 202,133 | -5% | 201,445 | -0.3% |
| nominal | 50 | 6 | 200,000 | 53,251 | -10% | 53,914 | 1.2% |
| nominal | 50 | 6 | 500,000 | 72,361 | -10% | 73,449 | 1.5% |
| nominal | 50 | 6 | 1,000,000 | 101,295 | -10% | 103,512 | 2.2% |
| nominal | 50 | 6 | 2,000,000 | 159,083 | -10% | 158,553 | -0.3% |
| nominal | 67 | 1.5 | 200,000 | 32,336 | 0% | - | - |
| nominal | 67 | 1.5 | 500,000 | 48,760 | 0% | - | - |
| nominal | 67 | 1.5 | 1,000,000 | 74,757 | 0% | - | - |
| nominal | 67 | 1.5 | 2,000,000 | 125,094 | 0% | - | - |
| nominal | 67 | 1.5 | 5,000,000 | 265,831 | 0% | - | - |
| nominal | 67 | 3 | 200,000 | 31,656 | -5% | 31,666 | 0.0% |
| nominal | 67 | 3 | 500,000 | 46,755 | -5% | 46,859 | 0.2% |
| nominal | 67 | 3 | 1,000,000 | 70,920 | -5% | 71,030 | 0.2% |
| nominal | 67 | 3 | 2,000,000 | 117,127 | -5% | 117,042 | -0.1% |
| nominal | 67 | 3 | 5,000,000 | 247,384 | -5% | 248,020 | 0.3% |
| nominal | 67 | 6 | 200,000 | 30,437 | -10% | 30,450 | 0.0% |
| nominal | 67 | 6 | 500,000 | 43,931 | -10% | 44,179 | 0.6% |
| nominal | 67 | 6 | 1,000,000 | 65,813 | -10% | 66,274 | 0.7% |
| nominal | 67 | 6 | 2,000,000 | 107,517 | -10% | 108,214 | 0.6% |
| nominal | 67 | 6 | 5,000,000 | 223,525 | -10% | 226,068 | 1.1% |

Table 7: Certainty equivalent performance improvement after correcting for bias for a number of rich stock and bond model scenarios with SPIAs.

401(k) is available to them. They are in good health, with a life expectancy 3 years longer than the average for their age. Their relative risk aversion is 6.

A naive application of the 4% rule in a Monte Carlo simulator using a 60/40 asset allocation, nominal bonds, and no SPIAs, would have them achieving a CE of $37,097 in retirement. They can do better using a 3% rule: $40,455.

One small technical detail is that for the percent rule consumption is capped at 30% of current period income plus estimated investment assets after adjusting for future taxes. Without this cap it is possible to consume the entire remaining investment portfolio in one time period, and then not have enough guaranteed income to pay the resulting taxes due the next period. This would result in consumption of zero in that period, which is infinitely bad, and produces a CE of 0.

In our naive application of the percent rule we used IID returns, but overlooked the fact that the values used in the Monte Carlo simulation should come with standard error. After factoring in the standard error things look worse. We get a CE of $34,762 for the 4% rule and $37,328 for the 3% rule.

Things improve though once we include the full stock and bond models which exhibit mean reversion. Using 20 years for the bond duration, the 4% rule has a CE of $36,522, and the 3% rule has a CE of $39,776.

The percent rule isn't a very good rule in that it isn't responsive to random changes in portfolio size after the age of retirement, and is dependent on portfolio history. Rules that are responsive to portfolio size and remaining life expectancy should be expected to do much better. PMT based rules, named after the Excel PMT function, are a popular alternative. The PMT function returns the payout amount required to deplete the portfolio over some remaining life expectancy assuming some fixed rate of return. I chose to benchmark against them because they appear to outperform other similar approaches.

The PMT based approach has been codified on the Bogleheads forum as Variable Percentage Withdrawal (VPW). VPW with a 30% consumption cap yields a CE of $48,575. If you annuitize, then VPW also recommends applying a consumption cap of 10%. Applying this consumption cap now boosts performance to $51,362. These values are quite a bit better, but we can improve further.

If we purchase annuities at age 80, as suggested by VPW, the CE jumps to $55,653. VPW suggests annuitizing enough that you can live comfortably, which is difficult to operationalize. Instead I chose to annuitize 30% of the investment portfolio, which through trial and error I discovered was the optimal fixed fraction to annuitize.

Switching from nominal to real bonds further boosts the CE to $59,561. This is good.

Now time to try AIPlanner. For this scenario AIPlanner delivered a CE of $79,924 before correcting for expected bias, and $81,449 after correcting for expected bias. This represents a major improvement over the other approaches. AIPlanner outperformed VPW with real bonds and SPIAs by 37%. The VPW scheme doesn't recommend the use of real bonds. If the client was previously using nominal bonds and SPIAs, then the switch to AIPlanner, real bonds, and SPIAs yields a gain of 46%.

A few notes. First, PMT based schemes are capable of delivering a CE of up to $71,378. This is for the use of a dynamically varying life expectancy rather than assuming you will live to 100 as in VPW, an assumed rate of return of 0.0% rather than the VPW default value of 3.8% for a 60/40 asset allocation, a 30% cap of consumption, annuitization 70% of the investment portfolio at age 80, an asset allocation of 60/40 before annuitization and 100/0 afterwards, and real bonds. Some of these values are VPW defaults, but other VPW defaults are far from optimal. Determining the optimal values for this scenario was through a trial and error process involving perhaps 50 simulations, each taking about 10 minutes. The optimal values to use will be different for each financial scenario. And that is the issue. This is a very time consuming approach that delivers sub-par results. AIPlanner delivers 14% better results almost instantly.

Second, AIPlanner blinded to the stock price, stock volatility, and the real interest, rate delivered without bias correction a CE of $78,452. When AIPlanner is able to observe these parameters again without bias correction it delivered a CE of $79,924. This means that AIPlanner is taking advantage of the stock price, volatility, and real interest rate, but the gains it is able to eek out from doing so are small. This points to the difficulty of tactical asset allocation and market timing in general. The 20% standard deviation stock price to fair value noise factor which represents the imprecision in determining if stocks are over or undervalued significantly reduces the signal present. Without noise for every 10% stocks are over valued there is a 1% drop in expected return. Measurements show that with noise applied for every 10% stocks appear to be overvalued there is only a 0.6% drop in expected return. Stock market volatility operates over time periods that are typically less than a year. Since AIPlanner currently operates with a 1 year time period this doesn't rule out the possibility of market timing volatility.

### 3.7 How Much Room is there for Further Improvement?

Unlike stochastic dynamic programming, AIPlanner doesn't compute the optimal solution. Instead it is a heuristic tool that hopefully delivers a near optimal solution. How close, we don't know, but the performance of AIPlanner on the log-normal IID models provides one estimate.

It is possible to train a version of AIPlanner for a specific scenario to get a sense of how much we might be leaving on the table by using a family of pre-trained generic models rather than training specific models for each scenario.

For the previous age 30, no assets scenario, AIPlanner delivered a bias corrected CE of $81,449. A specifically trained model for this scenario along with the optimally chosen bias achieved a CE of $82,825. An improvement of 1.7%.

What about a more complex scenario? For this I boosted the age 30 scenario pre-tax wages to $120k and added a bunch of liabilities. A nominal $5k per year student loan with 20 years remaining. An inflation adjusted $10k per year to support a 5 year old child until they turn 18. An inflation adjusted $30k per year of subsequent college expenses for 4 years. An inflation adjusted $50k home down payment and a nominal $20k per year 30 year mortgage starting at age 35.[4] In this case AIPlanner delivers a bias corrected CE of $87,235 versus $88,677 for the specifically trained model. Again a difference of 1.7%. Handling the more complex scenario appears to impose little performance cost.

## 4 Discussion

Reinforcement learning has proven successful at performing multi scenario financial planning. The key to doing this was simply increasing the number of timesteps for which the model is trained from 2 million to 100 million.

### 4.1 Risk Aversion

Previously AIPlanner was only evaluated with a risk aversion of 3. It is pleasing to see reinforcement learning is capable of handling the harder problem of a risk aversion of 6.

Single time period optimization, such as Modern Portfolio Theory (MPT), produces results that depend on the degree of risk to be taken. Unfortunately MPT is silent on the important question of how much risk to take, and the amount of risk to take will vary from time period to time period, and as a function of portfolio size. Lifecycle finance, or multi-period optimization, typically assumes a constant relative risk

[4] The $20k of Social Security per year starting at age 67 is also made taxable, correcting an earlier oversight.

| RRA | extra consumption | consumption uncertainty |
|---|---|---|
| $\infty$ | 0 | 0 |
| 6 | 27% | 19% |
| 3 | 68% | 40% |
| 1.5 | 226% | 90% |

Table 8: Consequences of different levels of relative risk aversion for a 20 year time period. Consumption uncertainty is defined here as the standard deviation of consumption divided by the mean consumption level.

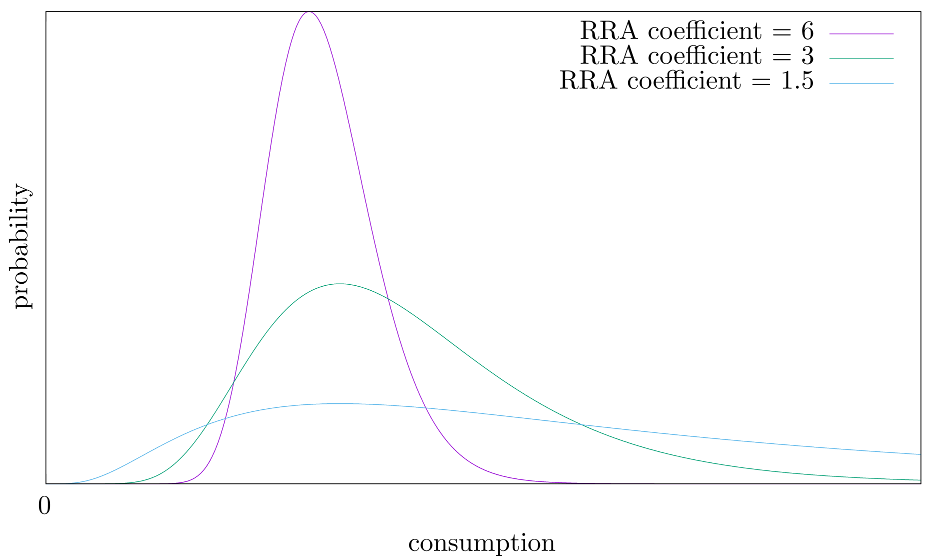


Figure 3: Consumption distributions for different levels of relative risk aversion.

aversion, which boils down the complicated question of how much risk to take to a single number that is invariant from time period to time period and also as a function of portfolio size.

Risk aversion is a personal choice. It can be surprisingly difficult for an individual to determine their risk aversion. One way to approach the problem is to present the individual with tables and graphs showing the consequences of different levels of risk aversion as shown for a 20 year time period in Table 8 and Figure 3. A 20 year time period seems appropriate. Assuming an individual doesn't start saving for retirement in a major way until around age 50 and retires at age 67, then they will have saved roughly half their investment portfolio by about age 60, and might be expected to have depleted half their investment portfolio by age 80. This makes 20 years a rough estimate of the average length of time of assets in their portfolio.

## 4.2 Accuracy of Results

As an academic exercise to test the applicability of reinforcement learning to financial planning, whether the financial model contains bugs isn't very important. Reinforcement learning optimizes in the presence of any bug, evaluation evaluates in the presence of the same bug, and then different results all with the same bug are compared. In most cases the impact of the bug will cancel out.

As a tool in the real world AIPlanner is no good if it produces inaccurate results due to bugs. The fact that for log-normal IID returns in the absence of taxes AIPlanner produces close to the optimal theoretical results suggests that in the absence of taxes AIPlanner is free of major bugs. The tax handling code is complex though. Experiments show that tax free investment assets outperform the same amount of taxable investment assets which outperform tax deferred assets, and that taxable assets with an initial cost basis of 100% outperform taxable assets with an initial cost basis of zero. This is as one should expect. Traces of what is happening from one time period to the next also lend confidence in the accuracy of AIPlanner, but these traces haven't so far been studied to the level of detail that one can be fully confident in the results of AIPlanner. There could still be small bugs to be found.

Ideally it would be possible to cross validate the results of AIPlanner with another reinforcement learning financial planner implementation, or traces from a tax aware portfolio simulator.

### 4.3 Real bonds and SPIAs

My work with AIPlanner reaffirms two commonly underappreciated tools in the financial planning tool box: real bonds and SPIAs. Real bonds, or TIPS, remove inflation risk at negligible cost. There is no rational reason not to use them in place of nominal bonds. SPIAs reduce or eliminate longevity risk. If you are in good health and don't have a strong bequest motive, once again there is no good reason not to use them.

### 4.4 Availability

AIPlanner is available online at www.aiplanner.com. The website allows you to input data about life expectancy, assets, liabilities, and goals in retirement, and then produces consumption and asset allocation recommendations along with the results of a simulation using the AIPlanner recommendations.

Commercial use licenses for the AIPlanner technology are also available.

## 5 Conclusion

Reinforcement learning is capable of generating very high quality financial plans for complicated scenarios involving guaranteed income, rich stock and bond models, assets, liabilities, and taxes, for pre and post retirement scenarios. No other tool or technique comes close to being able to do this. Often the plans come within a few percent of the theoretical optimal result. For one reasonable lifecycle scenario following the reinforcement learning recommendations was found beat most other strategies by a wide margin, and the best discovered alternative by 14%. In the past financial planning was constrained by the complexity of the financial planning problem. This no longer appears to be the case. The biggest issue in applying reinforcement learning today might be in determining the accuracy of the financial model used compared to the real world.

# References


[Dimson et al., 2019] Dimson, E., Marsh, P., and Staunton, M. (2019). *Credit Suisse Global Investment Returns Yearbook 2019*. Credit Suisse, Zurich.

[Espeholt et al., 2018] Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I., Legg, S., and Kavukcuoglu, K. (2018). Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. https://arxiv.org/abs/1802.01561.

[Irlam, 2018] Irlam, G. (2018). Financial planning via deep reinforcement learning AI. http://ssrn.com/abstract=3201703.

[Kolm and Ritter, 2019] Kolm, P. N. and Ritter, G. (2019). Modern perspective on reinforcement learning in finance. http://ssrn.com/abstract=3449401.

[Liang et al., 2018] Liang, E., Liaw, R., Nishihara, R., Moritz, P., Fox, R., Goldberg, K., Gonzalez, J. E., Jordan, M. I., and Stoica, I. (2018). RLlib: Abstractions for distributed reinforcement learning. https://arxiv.org/abs/1712.09381.

[Schulman et al., 2017] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. https://arxiv.org/abs/1707.06347.

[Shiller, 2015] Shiller, R. J. (2015). *Irrational Exuberance: Revised and Expanded Third Edition*. Princeton University Press.

[Sutton and Barto, 2018] Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

[Wikipedia contributors, 2019a] Wikipedia contributors (2019a). Autoregressive conditional heteroskedasticity — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Autoregressive_conditional_heteroskedasticity&oldid=922010061. [Online; accessed 30-November-2019].

[Wikipedia contributors, 2019b] Wikipedia contributors (2019b). Heath-Jarrow-Morton framework — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Heath%E2%80%93Jarrow%E2%80%93Morton_framework&oldid=928448723. [Online; accessed 30-November-2019].

[Wikipedia contributors, 2019c] Wikipedia contributors (2019c). Hull–White model — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Hull%E2%80%93White_model&oldid=910299970. [Online; accessed 30-November-2019].