Optimal Asset-Liability Management for Insurers

XU YUNPENG, Zhang Jinggong[1]

[1] *Nanyang Technological University, Singapore*

Abstract

Insurers' asset portfolios are crucial to company-growth and liability management. In previous works, under the framework of time-consistent planning, such portfolio optimization problems are usually done by assuming Markovian processes for asset movements, formulating the problem, finding a closed-form solution of the optimal control to the problem under simplistic assumptions, constructing a neural network approximation of the solution under realistic assumptions, and comparing the difference between the neural network output and the output of closed-form solution given some testing data. In this work, we assume the asset portfolio consists of stock and bonds, the liability consists of claim amounts, and they are modeled using VARIMA time series. Optimal asset allocation and dividend payout rates (optimal control laws) of an insurer will be found using a feedforward neural network surrogate, without a closed-form solution. We will then prove that the surrogate serves as a valid approximation of the optimal control law without empirically comparing it to a closed-form solution.

Keywords: Time-Consistent Planning, Dynamic Programming, Neural Network Surrogate

1. INTRODUCTION

Insurers need investment porfolio optimisations in order to manage solvency issues by increasing the portolio value, and to improve company-level P/L figures by distributing dividends. It is important for an insurer to be able to find the optimal portfolio trading strategy and dividend payout strategies.

In this work, superscript denotes the label, and subscript denotes the index set of time. Right superscript of a term in brackets denotes the exponent. A lower-case term denotes a constant term, unless been specified as a control variable, and an upper-case term (excluding T, U, B) denotes a random variable, process, or a set. A lower-case bolded term denotes a vector, and an upper-case bolded term denotes a matrix. Vectors are assumed to be vertical, and terms listed horizontally inside brackets constitute a tuple. A subscript with a square bracket denotes the elements in the vector or tuple.

For this research, We plan to find the optimal control for an insurer that maximizes the utility of periodic dividend payouts and the terminal portfolio value.

2. PROBLEM SETUP

Let $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{P})$ be a discrete time filtered probability space with a finite discrete index set $N = \{0 < 1 < ... < T\}$ as time points when the controls are executed. Assume there exist m number of assets in a desired portfolio, in which one is government bond, and the market is complete and free of arbitrage. Let S_t^1 denote the price of the risk-free asset at time $t \in N$. Let L_t be the dollar claim amount to be paid at t. Let S_t^i, $i = 2, 3, ..., m$ be the risky assets' price at time t. Assume the vector of all risky assets, the risk-free asset, as well as the liability follows a VARIMA(p,d,q) process, where matrix parameters $\mathbf{\Phi}, \mathbf{\Psi}, \mathbf{\Sigma}$ establish the dependence between bond, stock prices and claim amounts, and the $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ introduces the randomness. We assume non-negative asset prices and liabilities, therefore, define

$$\mathbf{x}_t := (S_t^1, S_t^2, \dots, S_t^m, L_t)^\intercal \text{ where } (\Delta)^d \ln \mathbf{x}_t = \boldsymbol{\mu} + \sum_{j=1}^{p} \mathbf{\Phi}^j \cdot ((\Delta)^d \ln \mathbf{x}_{t-j+1} - \boldsymbol{\mu}) + \sum_{j=1}^{q} \mathbf{\Psi}^j \cdot \boldsymbol{\epsilon}_{t-j} + \boldsymbol{\epsilon}_t.$$

With $\mathcal{F}_0 = \sigma(\{S_0^i \forall i, L_0\})$ and $\mathcal{F}_t = \sigma(\{(\Delta)^d \ln S_t^i \forall i, (\Delta)^d \ln L_t, \epsilon_t, ..., S_0^i \forall i, L_0\})$, the above vector process above is $\mathcal{F}_t$ adapted.

On the construction of the insurer's asset portfolio, let the units of stock i in the portfolio at $t, t \in N$ be h_t^i (assume no short allowed). Define the portfolio value C_t at time t as below

$$C_t = \sum_{i=1}^{m} h_{t-1}^i \cdot S_t^i - L_t.$$

The previous number of shares is evaluated at the current asset price to arrive at the portfolio value, hence the insurer would experience the random change of asset price and claim amounts from $(\Delta)^d \ln \mathbf{x}_t$.

At each time point t, let the dividend payout rate be u_t^0, capped at a maximum dividend payout ratio r; let the constant premium rate received per time t be p, using the formulation of asset rebalance from the classical Agent's Problem, we construct the portfolio dynamics as below

$$\sum_{i=1}^{m} h_t^i \cdot S_t^i = C_t + p - u_t^0 \cdot C_t.$$

For the ease of calculations, we define the proportion of portfolio value of each stock in the portfolio as

$$u_t^i := \frac{h_t^i \cdot S_t^i}{C_t + p - u_t^0 \cdot C_t}, i = 1, 2, ..., m,$$

and the work thereafter treats the above dividend payout rate and proportion as the control variables, i.e. the admissible trading strategy in the below set

$$\mathcal{A}_t := \left\{ (\mathbf{u}_t)_{t=0,...,T-1} : \forall i \in \{0, 1, ..., m\}, u_t^i \in [0, 1 + (r-1)\mathbf{1}_{i=0}], \sum_{i=1}^{m} u_t^i = 1 \right\}$$

Deducting the claims occured at $t+1$, L_{t+1}, we are led to the following dynamics of portfolio value

$$C_{t+1} = \sum_{i=1}^{m} u_t^i \cdot R_{t+1}^i \cdot (C_t + p - u_t^0 \cdot C_t) - L_{t+1}$$

where $R_{t+1}^i = S_{t+1}^i / S_t^i$ is the return rate.

3. VALUE FUNCTION

The aim of the insurer is to maximise the utility from periodic dividend payouts as well as discounted terminal portfolio value at the discount rate of β. Assuming the insurer faces the utility function of the below form

$$U(\cdot) = \frac{1}{\gamma}(\cdot)^{\gamma}, 0 < \gamma < 1,$$

the second derivative of U is negative and therefore exhibits a risk-averse appetite toward dividend and terminal portfolio value.

Define $\mathbb{Y} = ([0, \infty))^{(m+1)\cdot(d+1)} \times (\mathbb{R})^{(m+1)\cdot(p+q)} \times \mathbb{R}$ as the state space of Y_t, where Y_t is the stochastic state variable, and can be realized as y,

$$y := ((B)^0 \ln \mathbf{x}, ..., (B)^d \ln \mathbf{x}, (B)^0 (\Delta)^d \ln \mathbf{x}, ..., (B)^{p-1} (\Delta)^d \ln \mathbf{x}, (B)^0 \boldsymbol{\epsilon}, ..., (B)^{q-1} \boldsymbol{\epsilon}, C).$$

with

$$\forall t, (B)^k X := X_{t-k} \big| \mathcal{F}_t, k = 0, 1, ...$$

where $(B)^k$ denotes the backward shift operator. Negative time points denote the realisations/observations of X before the time of evaluating the insurer's asset portfolio strategy.

Assume the insurer adopts constant risk discount rate β, the Markov decision problem that the insurer wishes to maximize at each t is formulated as below

$$\mathbb{E}\left(\sum_{s=t}^{T-1} U(u_s^0 \cdot C_s) + \beta^{T-t} \cdot U(C_T) \middle| Y_t = y \right), t \in N.$$

It follows that the value function is of the below form

$$V_t(y) = \sup_{(\mathbf{u}_s)_{s=t,\dots,T-1}\in\mathcal{A}_t} \mathbb{E}\Big(\sum_{s=t}^{T-1} U(u_s^0 \cdot C_s) + \beta^{T-t}\cdot U(C_T)\Big| Y_t = y\Big), t \in N.$$

Using the terminal condition at $t = T$, we have

$$V_T(y) = \frac{1}{\gamma}\cdot (C)^\gamma,$$

therefore by constructing the dynamic programming equation, we have

$$V_t(y) = \sup_{\forall \mathbf{u}_t\in\mathcal{A}_t}\Big\{\frac{1}{\gamma}\cdot (u_t^0\cdot C)^\gamma + \beta\cdot\mathbb{E}\Big(V_{t+1}(\sum_{i=1}^m u_t^i\cdot \tilde{R}^i\cdot (C+p-u_t^0\cdot C) - \tilde{L})\Big| Y_t = y\Big)\Big\}, t\leq T-1 \tag{1}$$

where

$$(\tilde{R}^1, \tilde{R}^2, \dots, \tilde{R}^m)^\intercal = \big((B)^{-1}\mathbf{x}_{[1:m]} \oslash \mathbf{x}_{[1:m]}\big), \tilde{L} = (B)^{-1}\mathbf{x}_{[m+1]}$$

$$(B)^{-1}\mathbf{x} = \exp\Big(\sum_{k=0}^d \binom{d}{k}(-1)^k (B)^k \ln \mathbf{x} + (B)^{-1}(\Delta)^d \ln \mathbf{x}\Big)$$

$$(B)^{-1}(\Delta)^d \ln\mathbf{x} = \boldsymbol{\mu} + \sum_{j=1}^p \boldsymbol{\Phi}^j\cdot (B)^{t-j+1}((\Delta)^d\ln\mathbf{x} - \boldsymbol{\mu}) + \sum_{j=1}^q \boldsymbol{\Psi}^j\cdot (B)^{t-j}\boldsymbol{\epsilon} + \boldsymbol{\epsilon}$$

B^{-1} here denotes the forward shift operator, as the opposite operation to B^1. Let $\hat{\mathbf{u}}_t$ be the optimal control law, for this problem, it is extremely time consuming to find the closed-form solution of $\hat{\mathbf{u}}_t$ and V_t. And we are led to the neural network surrogate of them.

4. SURROGATE

In this section, the above dynamic programming problem will be solved using a neural network surrogate. The general idea is adopted from [Tao Chen, Mike Ludkovski, and Moritz Voß (19 Dec 2023): On parametric optimal execution and machine learning surrogates, Quantitative Finance, DOI: 10.1080/14697688.2023.2282657]. The aim is to find the neural network that serves as approximations of the mappings of $V_{t+1} : N\times\mathbb{Y}\to\mathbb{R}, \forall t\leq T-2$. With the approximated mapping of V_{t+1}, for any sample path of $Y_t = y$, one standing at time t can replace the true V_{t+1} in Equation 1 by the approximated mapping of it, input y into the approximated mapping, and find the argmax of Equation 1 to find the value of the optimal controls for this sample path at t .

Assuming the future indeed follows the VARIMA model specifications, the first step is to generate n number of bounded training samples of $y\in\mathbb{Y}$, each is denoted as y^j, so that we have the training samples $Y^n\subseteq\mathbb{Y}$ of the state variables,

$$Y^n = (y^1, y^2, ..., y^n),$$

Those bounds are of careful choices that should cover almost all possible values in the next T time points and cannot be impractical (e.g. though liability in $(B)^0\ln\mathbf{x}$ can go close ∞, if T is chosen to be 5 months, then the bounds must be reasonably large numbers but not infinity). Here we choose the bounds for $y[1 : d+1+p+q]$ by simulating T time steps forward and observe the maximum and minimum values of each state variable. Based on the values of $y[1 : d+1+p+q]$ on each simulated path j, the lower bound of C can be determined by

$$\Big((1+\min_{\forall i,j}\{S_T^{i,j}/S_T^{i,j}\})\cdot ...\cdot\Big((1+\min_{\forall i,j}\{S_1^{i,j}/S_1^{i,j}\})\cdot(C_0+v^s\cdot p - r\cdot C_0) - \max_{\forall j}\{L_1^j\}\Big) - ... - \max_{\forall j}\{L_T^j\}\Big)$$

The upper bounds of C can be determined similarly.

$$\Big((1+\max_{\forall i,j}\{S_T^{i,j}/S_T^{i,j}\})\cdot ...\cdot\Big((1+\max_{\forall i,j}\{S_1^{i,j}/S_1^{i,j}\})\cdot(C_0+v^s\cdot p - 0\cdot C_0) - \min_{\forall j}\{L_1^j\}\Big) - ... - \min_{\forall j}\{L_T^j\}\Big)$$

Each element in training sample y^j is generated using sobol sequence sampling given that we obtained the bounds of state variables in $\mathbb{Y}$. Sobol sequence sampling provides a more uniform coverage than random sampling, and hence would prevent clustering of training samples in some regions of the input space $\mathbb{Y}$.

The second step is to define the one-step-forward transition of y given a control u and random noise $\boldsymbol{\epsilon}$ that acts similarly to the forward shift operator $(B)^{-1}$:

$$(\tilde{B})^{-1}\big(y, \boldsymbol{\epsilon}, \mathbf{u}\big) := \Big((B)^{-1}\mathbf{x}, ..., (B)^{d-2}\mathbf{x}, (B)^{-1}(\Delta)^d\mathbf{x}, ..., (B)^{p-2}(\Delta)^d\mathbf{x}, \boldsymbol{\epsilon}, ..., (B)^{q-2}\boldsymbol{\epsilon}, \sum_{i=1}^{m} u^i \cdot \tilde{R}^i \cdot (C + p - u^0 \cdot C) - \tilde{L}\Big)$$

The next step is to find the expectation component of the optimal value function numerically, we apply a quantization approach to do so. Given a level of quantization, let $\boldsymbol{w}$ denotes the weights vector of occurring $\boldsymbol{e}$ as a quantization of $\boldsymbol{\epsilon}$, we define $\mathring{V}_t(y)$ such that $\mathring{V}_t \simeq V_t$,

$$\mathring{V}_t(y) = \sup_{\forall \mathbf{u}_t}\left\{\frac{1}{\gamma} \cdot (u_t^0 \cdot C)^\gamma + \beta \cdot \boldsymbol{w}^\intercal \cdot \mathring{V}_{t+1}\big((\tilde{B})^{-1}(y, \boldsymbol{e_t}, \mathbf{u}_t)\big)\right\}, \text{ whenever } Y_t = y$$

$$\mathring{V}_T(y) = \frac{1}{\gamma} \cdot (C)^\gamma = V_T(y)$$

The final step is to find the approximation of $\mathring{V}_t(y)$ using backward iteration. Denote the feedforward neural network surrogate of the closed-form solution of V_t as $\mathring{V}^{\theta_t}$ where θ_t is the initialised neural network weights and biases. Then, define

$$v_t(y, \mathbf{u}) = \begin{cases} \frac{1}{\gamma} \cdot (\mathbf{u}_{[1]} \cdot C)^\gamma + \beta \cdot \boldsymbol{w}^\intercal \cdot \mathring{V}_{t+1}\big((\tilde{B})^{-1}(y, \boldsymbol{e_t}, \mathbf{u}_{[2:m+1]})\big), t = T-1, \\ \frac{1}{\gamma} \cdot (\mathbf{u}_{[1]} \cdot C)^\gamma + \beta \cdot \boldsymbol{w}^\intercal \cdot \mathring{V}^{\hat{\theta}_{t+1}}\big((\tilde{B})^{-1}(y, \boldsymbol{e_t}, \mathbf{u}_{[2:m+1]})\big), t < T-1, \end{cases}$$

where $\hat{\theta}$ will be introduced shortly. Given the closed-from expression of V_T, from $T-1$ to 1, for each $j = 1, ..., n$, we find the largest value of the function $v_t(y^j, \mathbf{u})$ using scipy.optimize, denoted as v_t^j. Then we train the optimal weights and biases $\hat{\theta}_t$ using those n number of samples of v_t^j corresponding to y^j such that the MSE is minimised, so the mapping of $\mathring{V}_t$ can be approximated by $\mathring{V}^{\hat{\theta}_{t+1}}$ and it is used in the next backward iteration.

Here, before the training of $\mathring{V}^{\theta_t}$, we allow it to be a function of y and $\mathring{V}^{\hat{\theta}_{t+1}}$; whereas after the training, the θ_t that minimises the MSE, i.e. the $\hat{\theta}_t$ is found, thus $\hat{\theta}_t$ is only a function of y. The entire procedure is summarised as below,

Algorithm 1 Construct surrogate

Generate Y^n that fills n locations in $\mathbb{Y}$.
Initialise `V_train` as an empty $(T-1) \times n$ space.
for $t = T-1, T-2, \ldots, 1$ **do**
  for $j = 1, 2, \ldots, n$ **do**
    Optimise $v_t(y^j, u)$ and find $\hat{v}_t^j$ using scipy.optimize.
    `V_train[t][j]` $= v_t^j$
  end for
  Use `V_train[t]` and Y^n to fit a mapping $\mathring{V}^{\hat{\theta}_t}$ as an approximation of V_t
end for

The construction of the above neural network is documented in (https://github.com/yyunpeng/NNsurrogate2). It adopts the idea of [Tao Chen, Mike Ludkovski, and Moritz Voß (19 Dec 2023): On parametric optimal execution and machine learning surrogates, Quantitative Finance, DOI: 10.1080/14697688.2023.2282657]. However, here we do not have the approximation of the optimal control as a function of state variables, and the value function will be directly used to find the value of optimal control for each sample path. That is, to find the optimal control of any sample path with state variables y, we utilise the definition of the value function and thus the optimal control is given by $\arg\max_{\forall \mathbf{u}_t \in \mathcal{A}_t}\{\frac{1}{\gamma} \cdot (u_t^0 \cdot C)^\gamma + \beta \cdot \boldsymbol{w}^\intercal \cdot \mathring{V}^{\hat{\theta}_t}\big((\tilde{B})^{-1}(y, \boldsymbol{e_t}, \mathbf{u}_t)\big)\}$. In this way, there is one less round of approximation, and for multiple output neurons, the MSE tends to be high during the training than that of the case with a single output neuron.

5. EXPERIMENTAL DESIGN

In this section, we demonstrate an experimental design of the optimal control yielded by the surrogate described in Section 4. The baseline assumptions of the experiment are as follows:

1. The life insurer is evaluating its asset portfolio strategy as of 2023-03-01, time units are in months, $T = 5$, and assumes the monthly claim and monthly close price of assets in the past 2 years follows a VARIMA(2,1,0) model.
2. The life insurer policy portfolio includes 1% of monthly provisional counts of deaths in the U.S. (https://data.cdc.gov/NCHS/Monthly-Provisional-Counts-of-Deaths-by-Select-Cau/9dzk-mvmi) based on the date range as specified in 1.. Sum assured is $10,000$ USD for every death and will be paid immediately at the end of the month of death. The sum assured is adjusted by monthly inflation rate (https://www.investing.com/economic-calendar/cpi-69). The insurer receives constant premium rate $p = 12,000$ USD (i.e. with risk loading of 1.2).
3. $m = 3$, assets to be traded are HYG (S^1), TIP (S^2) and VYM (S^3), corresponding the corporate bonds, government bonds and stock equities that typically appear in an insurer's asset portfolio. $r = 0.05$ and monthly risk discount rate $\beta = \frac{1}{1.06^{1/12}}$ are constant throughout, initial capital value $C_0 = 10,000,000$ USD. Asset price data are obtained from yahoo finance library `yfinance` based on the date range as specified in 1..

   Therefore, the insurer face the below VARIMA specifications, where parameters are obtained from the `VAR` library in `statsmodels.tsa.api`:

$$\mathbf{\Phi}^1 = \begin{pmatrix} -0.329690 & 0.027015 & 0.060717 & 1.236035 \\ 0.216525 & 0.104913 & -0.120489 & -0.348980 \\ -0.083926 & -0.161908 & -0.379740 & -0.311504 \\ 0.060125 & 0.037710 & 0.096742 & 0.046234 \end{pmatrix}, \mathbf{\Phi}^2 = \begin{pmatrix} 0.194874 & 0.278572 & 0.052002 & 1.352464 \\ -0.472776 & -0.589316 & -0.966508 & -1.167676 \\ -0.014314 & -0.003085 & -0.057754 & -0.562943 \\ 0.031239 & 0.026260 & 0.011696 & -0.345791 \end{pmatrix}$$

$$\boldsymbol{\mu} = \begin{pmatrix} 0.004610 \\ 0.003379 \\ 0.018692 \\ 0.001738 \end{pmatrix}, \mathbf{\Sigma} = \begin{pmatrix} 1.000000 & 0.776869 & 0.751031 & -0.048496 \\ 0.776869 & 1.000000 & 0.581434 & -0.073065 \\ 0.751031 & 0.581434 & 1.000000 & -0.107651 \\ -0.048496 & -0.073065 & -0.107651 & 1.000000 \end{pmatrix}$$

4. $n = 2^{11}$, quantization level is 1,000 to numerically compute the expectation component of the value function. Uniform quantization is adopted, therefore $\boldsymbol{w}^\intercal = (\frac{1}{1000}, ..., \frac{1}{1000})$. The
5. The life insurer must maintain a government bonds holdings of 20% to 30% of the portfolio value, and a corporate bonds holdings of 40% to 70% of the portfolio value, the rest shall go to stock equities investment. That is, the admissible strategies becomes

$$\mathcal{A}_t := \left\{ (\mathbf{u}_t)_{t=0,1,2,3,4} : u_t^0 \in [0, 0.05], u_t^1 \in [0.4, 0.7], u_t^2 \in [0.2, 0.3], u_t^3 = 1 - u_t^1 - u_t^2 \right\}$$

We adopted serveral benchmarking controls to evaluate the performance of the our optimal control. Portfolio allocations are calculated based on the proportion of corporate bond, government bond and stock equities that appear in the aggregate investment portfolio of life insurance companies summarised in annual reports of various insurance survey institutions:

1. The portfolio allocation is constant at 86.83%, 05.35%, 07.82% according to NAIC 2022.
2. The portfolio allocation is constant at 45.00%, 36.67%, 18.33% according to IAIS 2023.
3. The portfolio allocation is constant at 58.33%, 29.17%, 12.50% according to OECD 2023.
4. The portfolio allocation is constant at 60.00%, 25.00%, 15.00% according to Goldman Sachs 2023 recommendations.

5. The portfolio allocation is randomly chosen to be any of the above 4.

For all the above benchmarking strategies, the portfolio dynamic is the same as in Section 2, only the definition of $u_t^i, i = 1, 2, 3$ changes. Dividend payout strategy will be the same as the NN control. Testing is conducted by simulating 2^{11} sample paths of the state variables into the future T months, apply the NN and benchmarking controls, record the respective dividend payouts and terminal portfolio value, and compute the utility score

$$W^k := \sum_{t=0}^{T-1} \beta^t \cdot U(u_t^{0,k} \cdot C_t^k) + \beta^T \cdot U(C_T^k), k = NN, 1, 2, 3, 4, 5$$

for each sample path.

6. NUMERICAL RESULTS

The below tables show the empirical statistics coming from the tests:

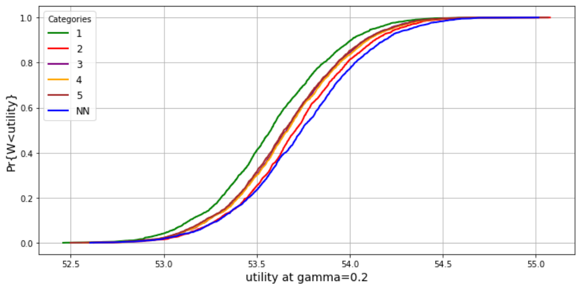
$\gamma = 0.2$				
k	$\mathbb{E}(W^k)$	$\text{std}(W^k)$	$\text{median}(W^k)$	$\Pr\{W^{NN} < W^k\}$
NN	53.74223	0.35030	53.74253	0
1	53.57918	0.33353	53.57594	0.01857
2	53.71229	0.33087	53.70948	0.32160
3	53.66275	0.32833	53.66146	0.14027
4	53.67002	0.33335	53.66812	0.14858
5	53.65608	0.33191	53.64634	0.13587

$\gamma = 0.4$				
k	$\mathbb{E}(W^k)$	$\text{std}(W^k)$	$\text{median}(W^k)$	$\Pr\{W^{NN} < W^k\}$
NN	62.69692	1.06040	62.69736	0
1	62.03475	0.98761	62.00824	0.01711
2	62.56425	0.97783	62.52849	0.30450
3	62.36660	0.96649	62.34267	0.13001
4	62.39574	0.98661	62.37523	0.13685
5	62.33985	0.98784	62.31906	0.12512

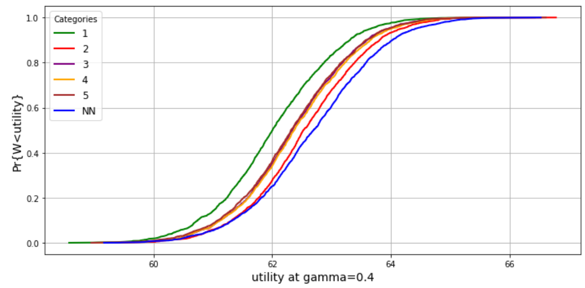
$\gamma = 0.6$				
k	$\mathbb{E}(W^k)$	$\text{std}(W^k)$	$\text{median}(W^k)$	$\Pr\{W^{NN} < W^k\}$
NN	125.96227	3.80895	125.98895	0
1	123.28073	3.48827	123.12844	0.01613
2	125.38710	3.46189	125.29005	0.29668
3	124.59852	3.41074	124.53219	0.12366
4	124.71540	3.49090	124.64409	0.13001
5	124.48828	3.47850	124.41996	0.11339

$\gamma = 0.8$				
k	$\mathbb{E}(W^k)$	$\text{std}(W^k)$	$\text{median}(W^k)$	$\Pr\{W^{NN} < W^k\}$
NN	335.34797	14.65791	335.23303	0
1	324.50697	13.26747	323.82632	0.01466
2	332.88720	13.23322	332.52485	0.28543
3	329.74052	13.00130	329.38590	0.11193
4	330.20937	13.32312	329.85156	0.11828
5	329.29787	13.23017	328.70783	0.10753

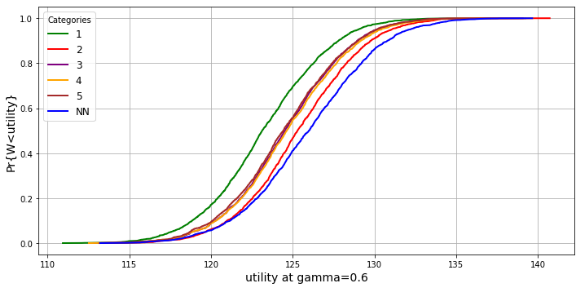
The plots below show the stochastic dominance of the NN control over the benchmarking controls.



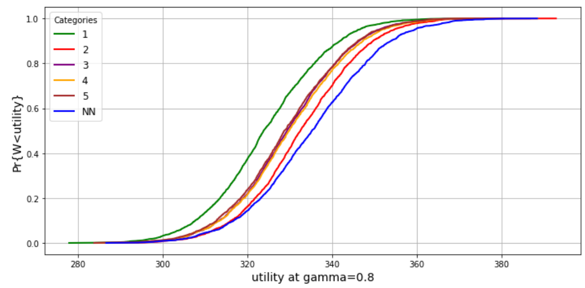
(a)



(b)



(c)



(d)

Figure 1: Plots of Empirical CDF of utility score with varying $\gamma = 0.2, 0.4, 0.6, 0.8$ for the NN and 5 benchmarking controls.