

Week3B Extra Credit

AUTHOR
Bryan Zhang

3. ****EXTRA CREDIT:** Read the papers “High Water Marks and Hedge Fund Management Contracts” by Goetzmann, Ingersoll, and Ross (The Journal of Finance, August 2003). Since this is a paper aimed at readers with Ph.D. degrees in finance or economics, please feel free to skim through equations that you do not understand, but instead focus on the intuition behind the authors' ideas and the content of the tables. Provide a brief summary of what you think this paper says that might be of interest to potential limited partners in hedge funds, such as Yale or Harvard.

The highwater mark refers to an incentive structure that hedge funds use. The idea is that they get paid a “bonus” that is calculated by a percentage of profits (the standard is 20%) above the previous high point in the fund. The authors of of the paper suggest that hedge funds are different from traditional mutual funds in that returns tend to diminish with size due to the underlying (arbitrage) strategies. Successful hedge funds are usually closed to new investors while the opposite is true for “successful” mutual funds. The most likely reason is that hedge fund strategies try to take advantage of “market opprotunities” that exploit small price discrepancies, and having higher amounts of capital to invest can mean greater price impact which would saturate the “market opprotunity”. The authors suggest that this is a contributor to why high water mark contracts exist, because hedge funds don't aim to grow their AUM, so they need other sources of income to survive. Another important aspect of hedge funds is that the volatility of the underlying portfolio can influence the amount of fees paid. A portfolio with higher volatility might have higher performance fees compared to a portfolio with lower volatility. The authors suggest that this may encourage hedge fund managers to increase the amount of risk in their strategies.

4. *****EXTRACREDIT(VERYHARD):**Trytoreplicateaspecialcaseofthethodologyinthe paper by Goetzmann, Ingersoll, and Ross.

In the code below I tried simulating the values of regular **annual** and **incentive** fees using the model provided for finding:

1. Total Fees $f(S, H)$
2. Performance Fees $p(S, H)$
3. Regular Fees $a(S, H)$

Note: The tables below replicate Table 1 in the paper

Some interesting observations:

- a higher standard deviation of returns σ increases the total fees when the probability of liquidation b is 0, but when the probability is higher than 0, a higher σ decreases the total fees
 - despite total fees falling, the performance fee increases when the probability of default is less than or equal to .5
- as the NAV of the hedge fund S declines relative to the high water mark H , the total amount of fees fall. I think this makes economic sense.
- as the probability of liquidation increase, the total fee amount decreases as well

```
import numpy as np
import pandas as pd

def find_roots(r, sigma, k, c, c_1, g, w, lmbda, b):
    first_term = .5 * sigma **2 + c - r - c_1 + g
    second_term = np.sqrt((.5 * sigma **2 + c - r - c_1 + g) ** 2 + (2 * sigma ** 2) *

    first_root = (first_term + second_term) / sigma ** 2 # gamma
    second_root = (first_term - second_term) / sigma ** 2 # n

    return first_root, second_root
```

```
def present_value_of_fees(S, H, r, sigma, k, c, c_1, g, w, lmbda, b, gamma, n):
    first_term = c / (c + w + lmbda)

    w2_num = k * (w + lmbda) + c * b ** (1 - n) * (n * (1 + k) - 1)
    w2_dem = (c + w + lmbda) * (gamma * (1 + k) - 1 - b ** (gamma - n) * (n * (1 + k) -
    second_term = w2_num/w2_dem * (S / H) ** (gamma - 1)

    w3_num = b ** (gamma - n) * k * (w + lmbda) + c * b ** (1 - n) * (gamma * (1 + k) -
    w3_dem = (c + w + lmbda) * (gamma * (1 + k) - 1 - b ** (gamma - n) * (n * (1 + k) -
    third_term = w3_num/w3_dem * (S / H) ** (n - 1)

    return first_term + second_term - third_term
```

```
def present_value_performance_fee(S_over_H, gamma, n, k, b):
    w1_num = (S_over_H ** (gamma - 1)) - (b ** (gamma - n)) * (S_over_H ** (n - 1))
    w1_den = gamma * (1 + k) - 1 - (b ** (gamma - n)) * (n * (1 + k) - 1)
    return (w1_num/w1_den) * k
```

```
# fixed parameters
r = .07
c = .015
c1 = .03
g = .05
k = .2
H = 1

# simulating fees
for b in [0, .5]:
    for sigma in [.15, .25]:
        for w in [.025, .05]:
            lmbda = w
            gamma, n = find_roots(r, sigma, k, c, c1, g, w, lmbda, b)
            performance_fee_list = []
            total_fee_list = []
            regular_fee_list = []
            S_over_H_list = []
            for S in np.arange(1.0, .4, -.1):
                S_over_H = S / H
                performance_fee = round(present_value_performance_fee(S_over_H, gamma,
                total_fee = round(present_value_of_fees(S, H, r, sigma, k, c, c1, g, w,
                regular_fee = total_fee - performance_fee

                peformance_fee_list.append(performance_fee)
                total_fee_list.append(total_fee)
                regular_fee_list.append(regular_fee)
                S_over_H_list.append(S_over_H)

            df_dict = {
                "S/H": S_over_H_list,
                "Regular Fee": regular_fee_list,
                "Performance Fee": performance_fee_list,
                "Total Fee": total_fee_list,
            }
            df = pd.DataFrame(df_dict)
            display(f"b = {b}")
            display(f"w + lmbda = {w + lmbda}")
            display(f"sigma = {sigma}")
            display(df)
```

'b = 0'

'w + lmbda = 0.05'

'sigma = 0.15'

	S/H	Regular Fee	Performance Fee	Total Fee
0	1.0	0.200	0.131	0.331
1	0.9	0.204	0.116	0.320
2	0.8	0.207	0.102	0.309
3	0.7	0.211	0.088	0.299
4	0.6	0.214	0.074	0.288
5	0.5	0.216	0.061	0.277

'b = 0'

'w + lmbda = 0.1'

'sigma = 0.15'

	S/H	Regular Fee	Performance Fee	Total Fee
0	1.0	0.119	0.087	0.206
1	0.9	0.120	0.073	0.193
2	0.8	0.123	0.059	0.182
3	0.7	0.124	0.047	0.171
4	0.6	0.126	0.036	0.162
5	0.5	0.127	0.026	0.153

'b = 0'

'w + lmbda = 0.05'

'sigma = 0.25'

	S/H	Regular Fee	Performance Fee	Total Fee
0	1.0	0.188	0.186	0.374
1	0.9	0.191	0.172	0.363
2	0.8	0.194	0.158	0.352
3	0.7	0.198	0.143	0.341
4	0.6	0.201	0.128	0.329
5	0.5	0.205	0.112	0.317

'b = 0'

'w + lmbda = 0.1'

'sigma = 0.25'

	S/H	Regular Fee	Performance Fee	Total Fee
0	1.0	0.114	0.128	0.242
1	0.9	0.116	0.114	0.230
2	0.8	0.117	0.100	0.217
3	0.7	0.119	0.086	0.205
4	0.6	0.121	0.072	0.193
5	0.5	0.122	0.059	0.181

'b = 0.5'

'w + lmbda = 0.05'

'sigma = 0.15'

	S/H	Regular Fee	Performance Fee	Total Fee
0	1.0	0.173	0.123	0.296
1	0.9	0.175	0.108	0.283
2	0.8	0.173	0.092	0.265
3	0.7	0.160	0.074	0.234
4	0.6	0.122	0.048	0.170
5	0.5	0.000	0.000	0.000

'b = 0.5'

'w + lmbda = 0.1'

'sigma = 0.15'

	S/H	Regular Fee	Performance Fee	Total Fee
0	1.0	0.111	0.085	0.196
1	0.9	0.111	0.071	0.182
2	0.8	0.110	0.057	0.167
3	0.7	0.104	0.042	0.146
4	0.6	0.081	0.026	0.107
5	0.5	0.000	0.000	0.000

'b = 0.5'

'w + lmbda = 0.05'

'sigma = 0.25'

	S/H	Regular Fee	Performance Fee	Total Fee
0	1.0	0.097	0.136	0.233
1	0.9	0.097	0.121	0.218
2	0.8	0.093	0.103	0.196
3	0.7	0.083	0.080	0.163
4	0.6	0.057	0.049	0.106
5	0.5	0.000	0.000	0.000

'b = 0.5'

'w + lmbda = 0.1'

'sigma = 0.25'

	S/H	Regular Fee	Performance Fee	Total Fee
0	1.0	0.074	0.110	0.184
1	0.9	0.074	0.095	0.169
2	0.8	0.072	0.078	0.150
3	0.7	0.064	0.059	0.123
4	0.6	0.044	0.036	0.080
5	0.5	0.000	0.000	0.000