Portfolio Managment: Homework 3

Gal Skarishevsky, Nathan Matare, Brian Thompson, Lior Sahaf

April 11, 2017

A.1. Questions on Treasury Inflation-Protected Securities (TIPS).

a)

Regular bonds pay a fixed coupon throughout their life and return their pre-established face value at maturity. TIPS adjust to inflation – the principal value is pinned to the CPI, and coupon amounts are calculated based on their pre-established interest rate and the adjusted principal value.

b)

If we expect inflation to rise we can short treasuries and use the proceeds to purchase TIPS, profiting on the difference between the treasury rate and the TIPS rate on the inflation-adjusted principal amount.

A.2. Questions on HMC's portfolio.

a)

b)

B.1. Relative performance of stocks and T-bills.

a)

Considering that an unskilled investor could choose a stock only strategy and be 'correct' (that is, select the higher return) 68.1% of the time, Claire's "skill" is unconvincing. Claire is 8% less skilled than an investor who simply chooses a stock only strategy.

```
length(which(data$stocks > data$tbills)) / NROW(data) * 100
```

```
[1] 68.1
```

b)

```
cumreturn <- (data + 1)
kable(t(apply(cumreturn, 2, prod)), digits = 6) # in dollars
```

tbills	stocks
20.5	4598

B.2. Perfect vs. random market timing.

a)

Naturally, the omniscient strategy performs better as it selects the highest return for each given year.

cumreturn_perfect	mean_perfect	sr_perfect	mean_market	sr_market
648122	0.166	0.964	0.117	0.405

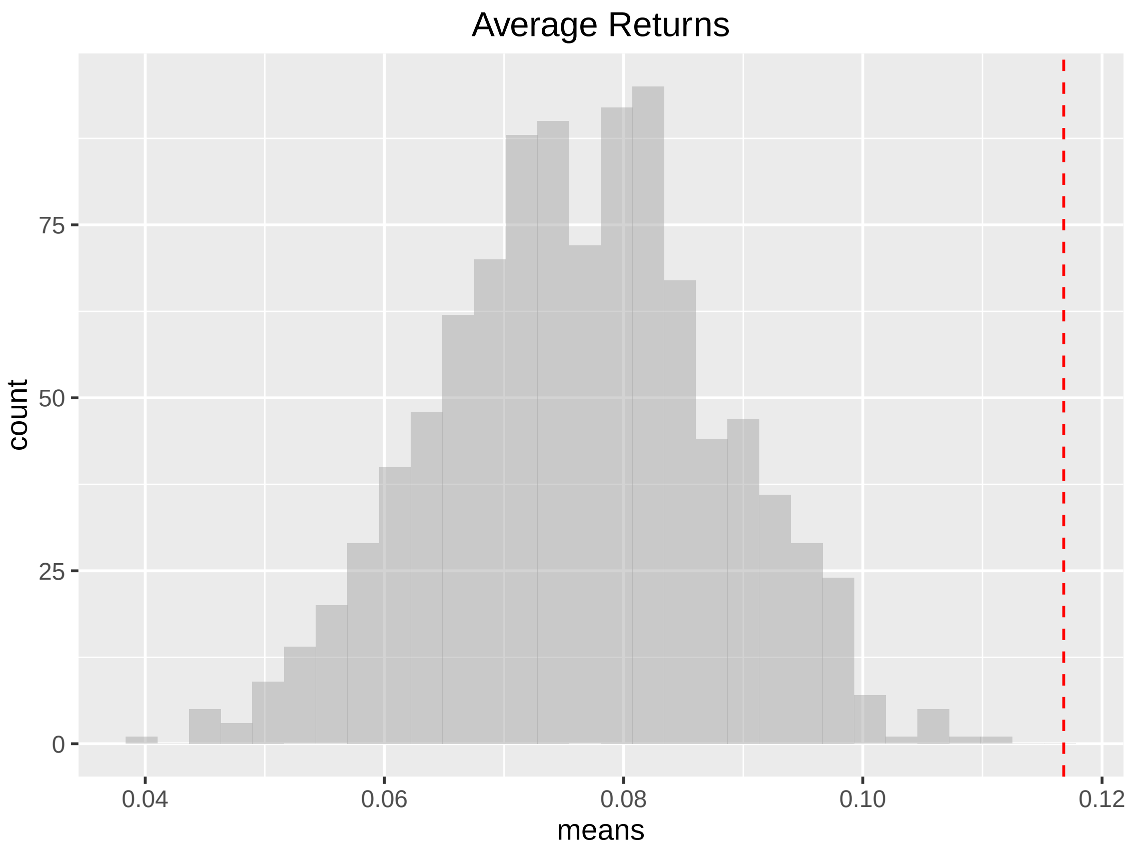
b)

We write the function '`randomTiming`' that randomly selects between the treasury bill and market.

```
randomTiming <- function() as.matrix(apply(data, 1, function(x) sample(x, 1)))
# run 1000 simulations of random timing
simulations <- replicate(1000, randomTiming(), simplify = FALSE)

sim_means <- unlist(lapply(simulations, mean))
sim_sharperatios <- unlist(lapply(simulations,
                          function(x) mean(x - data$tbills) / sd(x - data$tbills)))
```

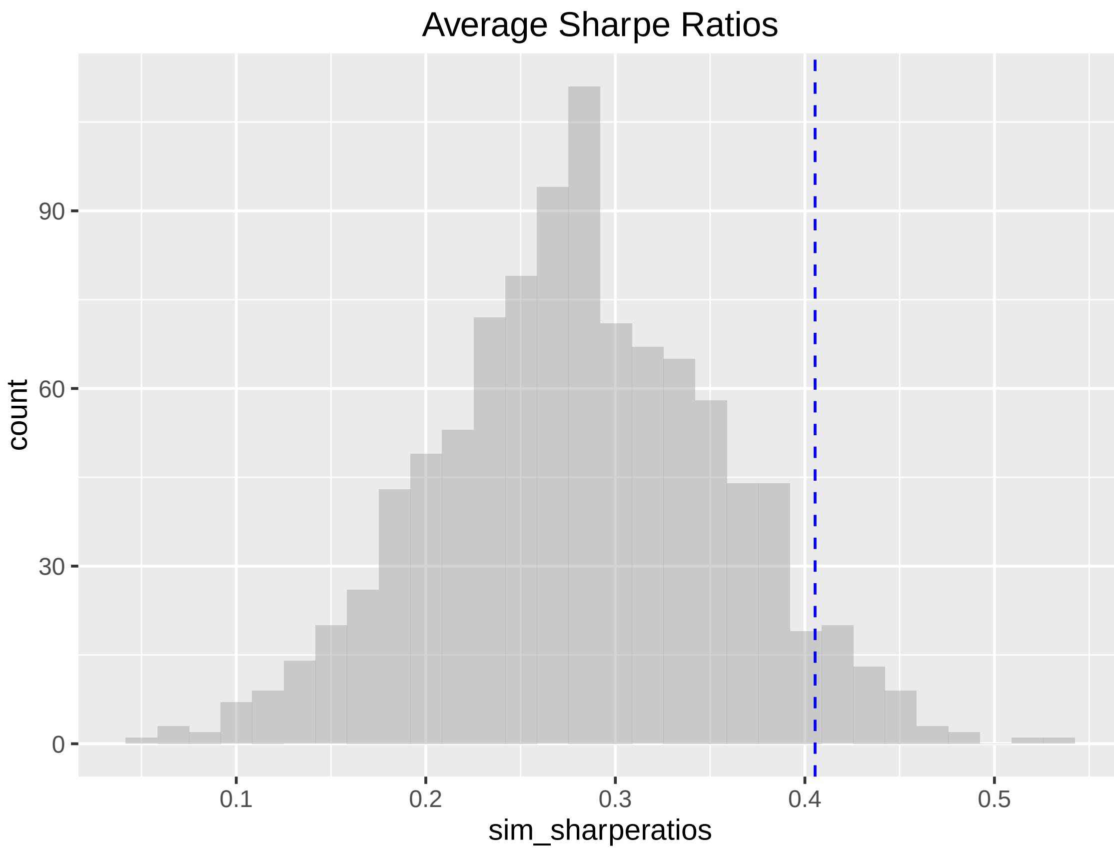
The expected return of Claire's random strategy is 7.62%. The average return of the stock only portfolio is shown in red.

```
[1] 0.0762
```



The expected Sharpe Ratio of Claire's random strategy is 0.28. The Sharpe Ratio of the stock only portfolio is shown in blue.

```
[1] 0.281
```



B.3. Benefits of imperfect market timing.

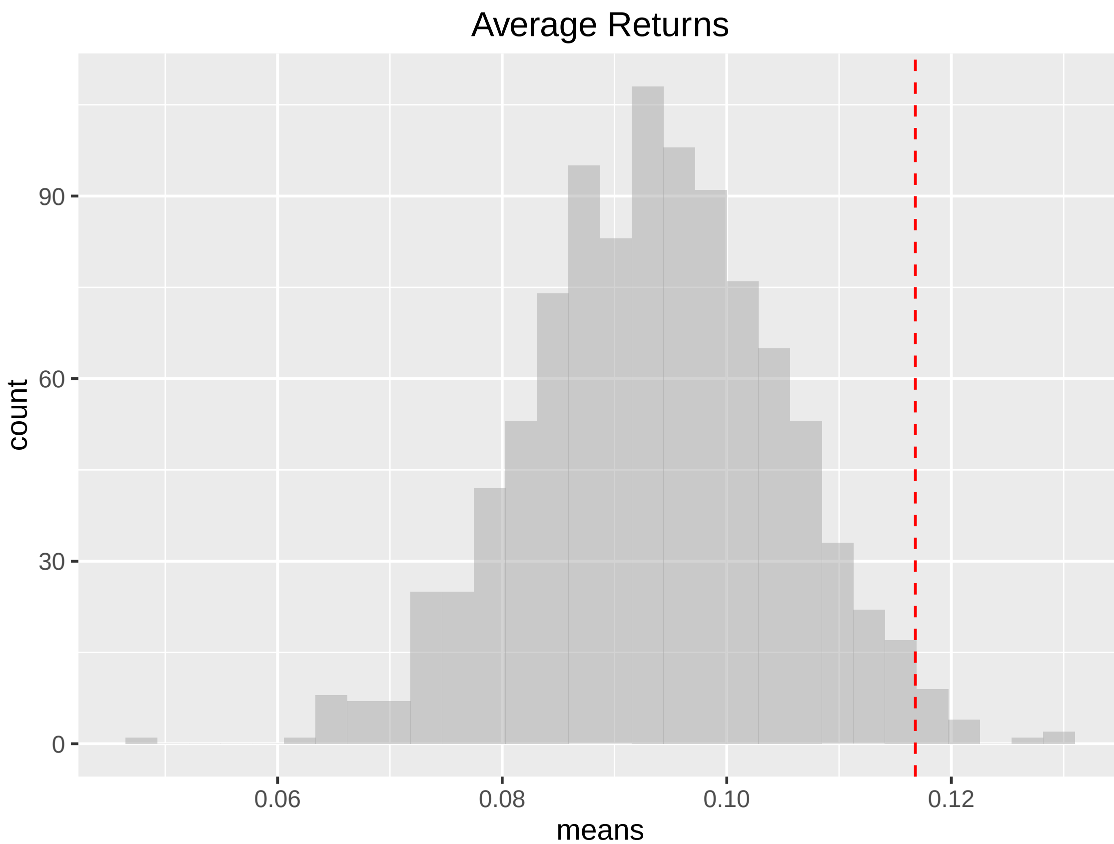
a)

We write the function '`skillTiming`' that selects the 'correct' secruity 60% of the time.

```
skillTiming <- function() as.matrix(apply(data, 1,
               function(x) sample(c(max(x), min(x)),
                                  prob = c(0.60, 0.40), 1)))
              # get the 'correct', aka max 60% of time
simulations <- replicate(1000, skillTiming(), simplify = FALSE)
# run 1000 simulations of random timing

sim_means <- unlist(lapply(simulations, mean))
sim_sharperatios <- unlist(lapply(simulations,
                           function(x) mean(x - data$tbills) / sd(x - data$tbills)))
```

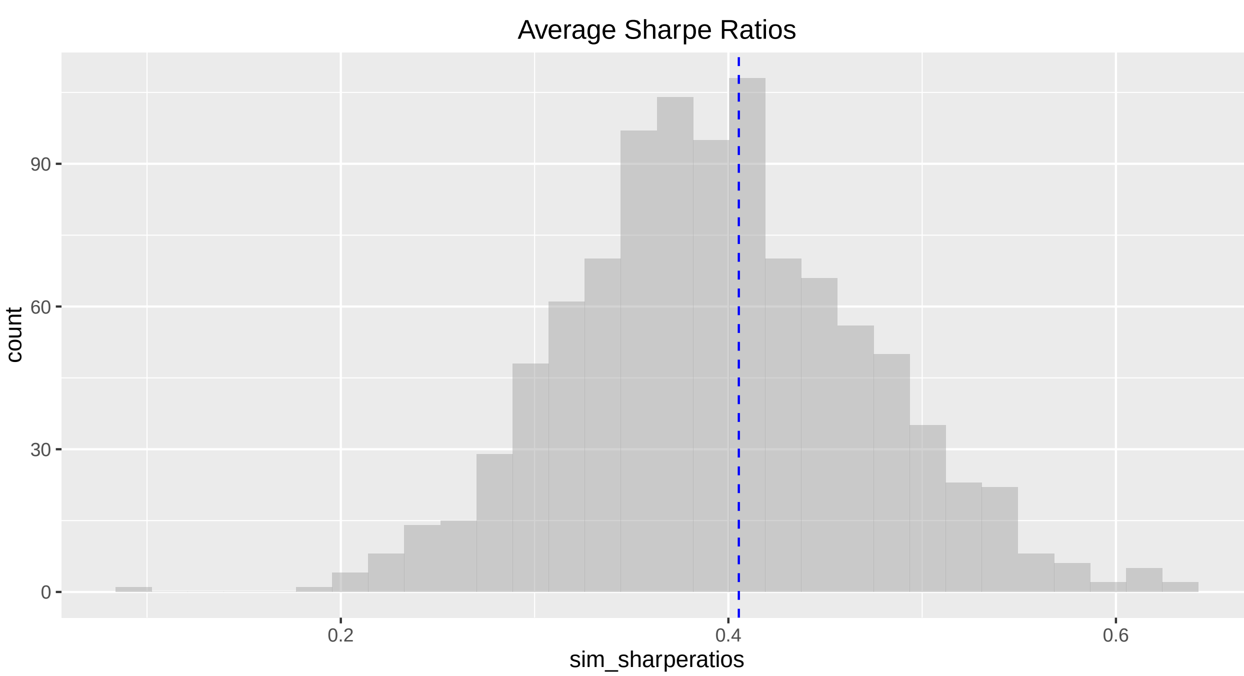
Claire's skill strategy yields an average return of 9.35%. Naturally, the skill strategy improves upon the random strategy but is still inferior to the market only portfolio.

```
[1] 0.0935
```



Equally, the 60% skill strategy yields an improved average Sharpe Ratio of 0.394.

```
[1] 0.394
```



b)

With 2% fees, Claire's skill strategy yields a 7.35% average return and 0.261 Sharpe Ratio. As this is no better than randomly selecting between stocks and treasury bills, and worse yet than selecting a market only portfolio, we would not hire Claire based upon this fee arrangement.

```
[1] 0.0735
```

```
[1] 0.261
```

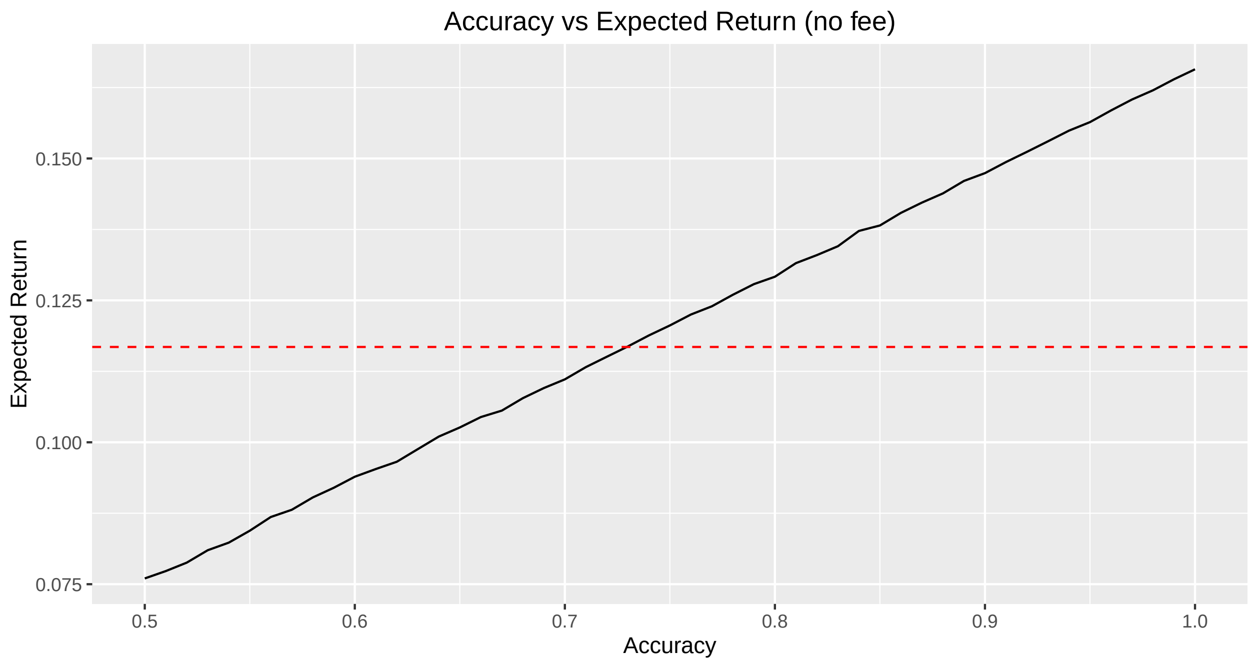
B.4. Imperfect market timing with different forecasting accuracies.

We write the function '`skillTimingn`' that selects the 'correct' secruity n percent of the time. Next, we simulate 1000 trails at an accuracy of [0.50 - 1.00], and parallize the operation over multiple cores.

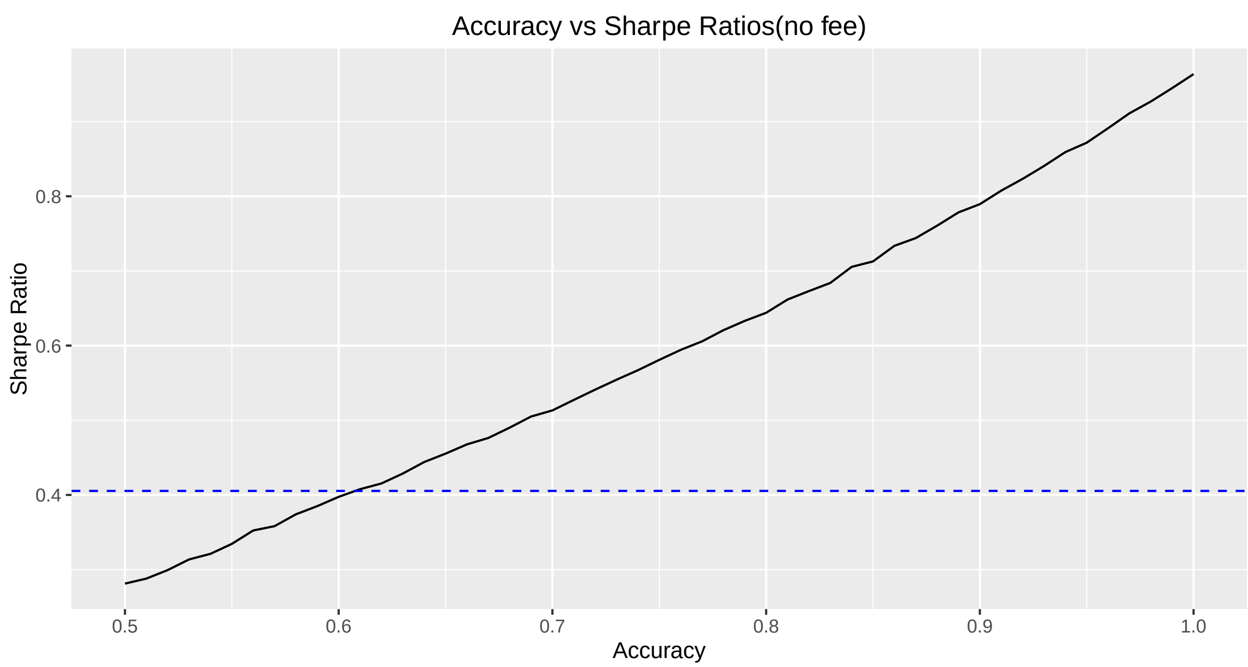
```
ns <- rep(50:100) / 100
skillTiming_n <- function(n) as.matrix(apply(data, 1,
                 function(x) sample(c(max(x), min(x)), prob = c(n, 1 - n), 1)))
# all(skillTiming_n(1.00) == perfect) # sanity check

registerDoMC(detectCores() - 1)
simulations <- foreach(i = ns) %dopar% {
  replicate(1000, skillTiming_n(i), simplify = FALSE)
  } # for each n, run a simulation

sim_means_ns <- unlist(lapply(simulations,
                      function(x) mean(unlist(lapply(x, mean)))))
sim_sharperatios <- unlist(lapply(simulations,
                      function(y) mean(unlist(lapply(y,
                      function(x) mean(x - data$tbills) / sd(x - data$tbills))))))
frame <- cbind.data.frame(ns, sim_means_ns, sim_sharperatios)
```

We plot the two cases: one without fees, and one with 2% management fees. To beat the stock-only strategy's expected return (11.68%), Claire requires a minimum accuracy of 73%. To exceed the market's Sharpe ratio (0.4054) requires an accuracy level of 61%. To beat the stock-only strategy's expected return (11.68%), Claire requires a minimum accuracy of 84%. To exceed the market's Sharpe ratio (0.4054) requires an accuracy level of 72%.

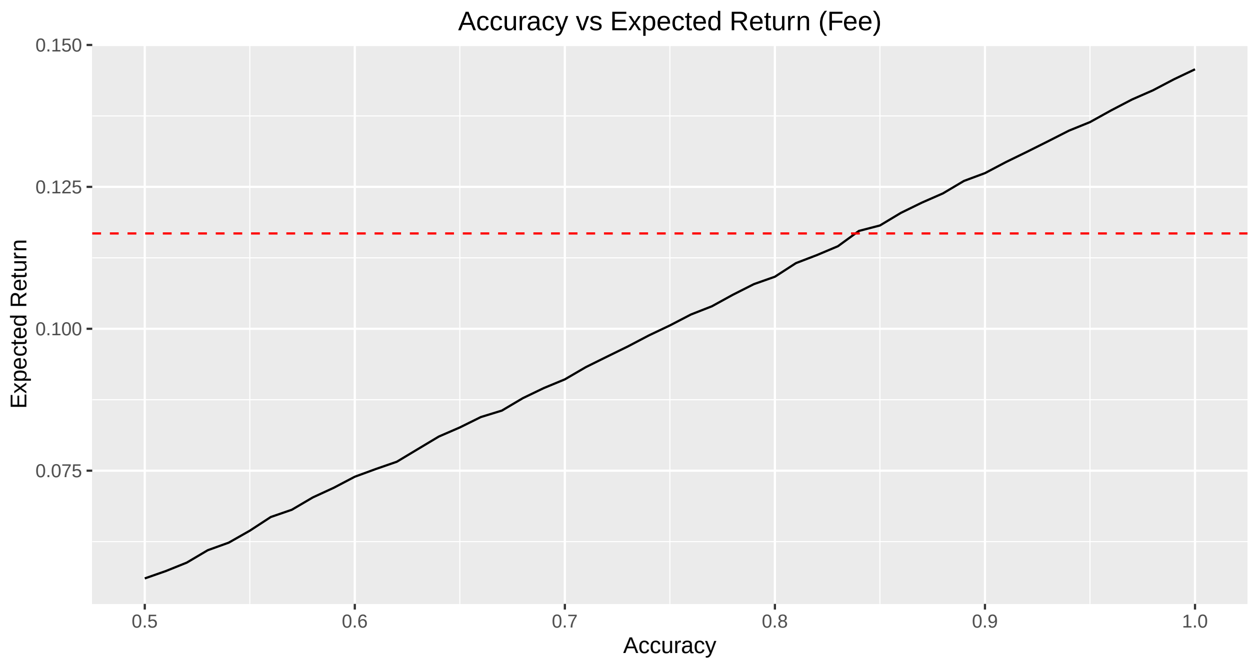
```
[1] 0.73
```


Accuracy vs Expected Return (no fee)
Expected Return
0.150
0.125
0.100
0.075
0.5
0.6
0.7
0.8
0.9
1.0
Accuracy

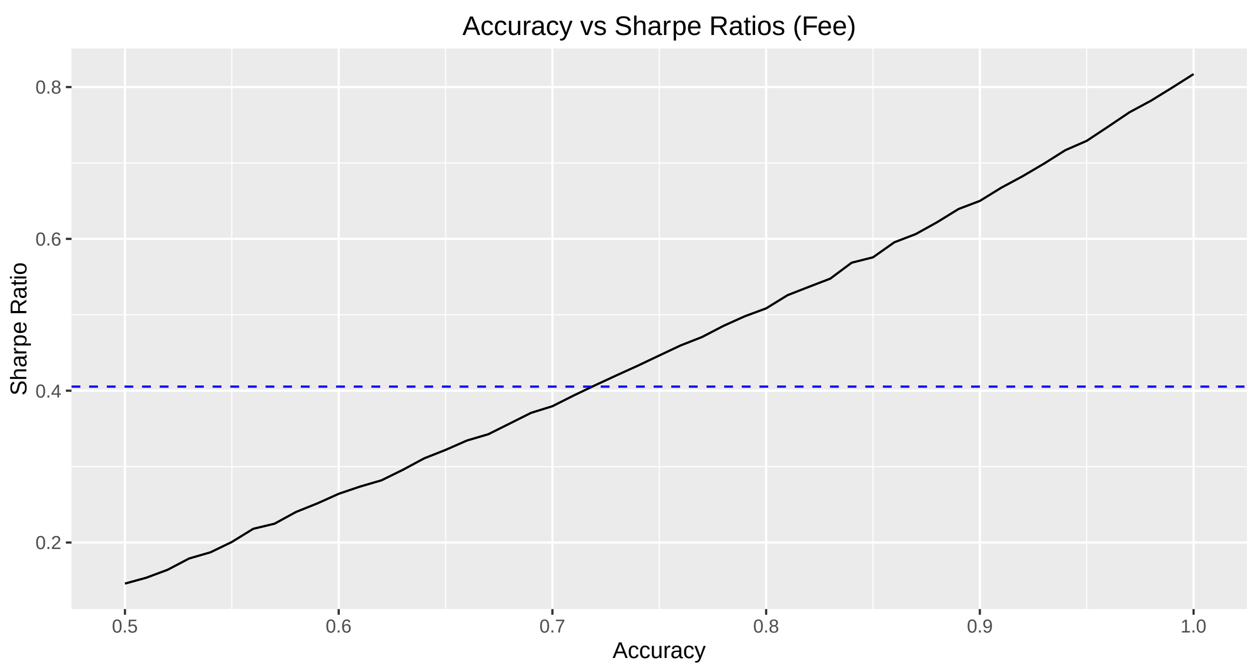
[1] 0.61


Accuracy vs Sharpe Ratios(no fee)
Sharpe Ratio
0.8
0.6
0.4
0.5
0.6
0.7
0.8
0.9
1.0
Accuracy

[1] 0.84


Accuracy vs Expected Return (Fee)
0.150
0.125
0.100
0.075
Expected Return
0.5
0.6
0.7
0.8
0.9
1.0
Accuracy

```
[1] 0.72
```


Accuracy vs Sharpe Ratios (Fee)
0.8
0.6
0.4
0.2
Sharpe Ratio
0.5
0.6
0.7
0.8
0.9
1.0
Accuracy