Jacobs University
Fall 2019

September 16, 2019

Stochastic Methods + Lab

Assignment Sheet 2

Due on September 30, 2019

Problem 1 [6 points]

Implement the bisection method, Newton's method, the secant method, and Brent's method as python functions for computing the IRR. Use `timeit` to compare the efficiency of these functions, using the parameters

```
N = 300
C = 120.0 * arange(10,N+10)
P = 15000
```

Problem 2 [2 points]

Jargon Jungle: The *yield to maturity* of a level coupon bond is the IRR of its cash flow. Compute the yield to maturity of a 10-year level coupon bond sold at 75% of par with a coupon rate of 10% paid semiannually.

Problem 3 [2 points]

Plot the price vs. time to maturity for level coupon bonds with annual coupon rates of 2%, 6%, and 12% paid semiannually. Assume a yield of 6% and a par value of 1 000 \$.

Problem 4 [2 points]

Consider a 10-year level coupon bond with an annual coupon rate of 8% compounded annually and a par value of 1 000 \$. Plot the price of the bond vs. the yield.

Problem 5 [3 points]

Plot price volatility vs. time to maturity for level coupon bonds with annual coupon rates of 2%, 6%, and 12% paid semiannually. Assume a yield of 6% and a par value of 1 000 \$. To see the different volatility behaviors, take a range from 0 up to 100 years to maturity.

Problem 6 [3 points]

Plot the bond value (forward value) of an 8% 15-year bond compounded semi-annually vs. years to maturity under three rate scenarios: (a) the interest rate decreases instantaneously to 6%, (b) the interest rate remains unchanged, and (c) the interest rate increases instantaneously to 10%.

Note: The forward value of the bond required here is the value of the bond at time zero based on the new market interest rate times the compounded interest factors corresponding to different times to maturity.

Problem 7 [2 points]

Plot the future value of a 30-year bond at a coupon rate of 10% compounded annually after a 10-year horizon as a function of yield. Find the minimum of the horizon price numerically, e.g., using `scipy.optimize.brent`.