FinTech Unit 4 Homework Grading Rubric

Criteria	Ratings – Max Points			
Data Preparation • Pandas is used to read each CSV file as a DataFrame • Null values have been detected and removed • Numeric values have been formatted and data types converted • S&P 500 closing prices have been converted to daily returns • Whale Returns, Algorithmic Returns, and the S&P 500 Returns are joined into a single DataFrame with columns for each portfolio's returns	**20 Points - Mastery** • Completed 5 out of 5 requirements • Code runs without error and produces the assigned results • Code accounts for all possible scenario • Code is free of bugs	**15 Points - Approaching Mastery** • Completed 3 out of 5 of requirements • Code runs without error • Code produces results as expected 80% or more of the time	**10 Points - Progressing** • Completed 2 out of 5 requirements • Code runs without error • Code produces results, but not necessarily the correct results	**5 Points - Emerging** • Completed 1 or none out of the 5 requirements • No submission • Code runs with error
Quantitative Analysis <u>Performance Analysis</u>: • Cumulative returns for each portfolio calculated and plotted. • Compares each portfolio's returns to the S&P 500 <u>Risk Analysis</u>: • Box plot for each of the returns created, with largest and smallest spread identified • Standard deviation is calculated for each portfolio, identify risk in comparison to S&P 500	**20 Points - Mastery** • Completed 8 out of 8 requirements • Code runs without error and produces the assigned results • Code accounts for all possible scenario • Code is free of bugs	**15 Points - Approaching Mastery** • Completed 5 out of 8 requirements • Code runs without error • Code produces results as expected 80% or more of the time	**10 Points - Progressing** • Completed 3 out of 8 requirements • Code runs without error • Code produces results, but not necessarily the correct results	**5 Points - Emerging** • Completed 2 or fewer out of the 8 requirements • No submission • Code runs with error

Rolling Statistics: • Rolling standard deviation of each portfolio is plotted against rolling standard deviation of S&P 500 • Risk of each portfolio is identified compared to S&P 500 risk • Correlation table for the algorithmic, whale and S&P 500 returns is included • Rolling beta of portfolios plotted against rolling beta of S&P 500				
Sharpe Ratios • Daily returns are used to calculate Sharpe Ratio • Sharpe ratios are visualized using a bar plot • Determines whether the algorithmic strategies outperform the whales or the S&P 500	**15 Points - Mastery** • Completed 3 out of 3 requirements • Code Runs without error and produces the assigned results • Code accounts for all possible scenario • Code is free of bugs	**10 Points - Approaching Mastery** • Completed 2 out of 3 requirements • Code runs without error • Code produces results as expected 80% or more of the time	**5 Points - Progressing** • Completed 1 out of 3 requirements • Code runs without error • Code produces results, but not necessarily the correct results	**0 Points - Emerging** • Completed none or partial out of the 3 requirements • No submission • Code runs with error
Custom Portfolio • Google Finance function is used to choose portfolio • Data downloaded as CSV files and portfolio returns calculated • Portfolio returns added to the DataFrame with the other portfolios analyzed and compared	**15 Points - Mastery** • Completed 3 out of 3 requirements • Code Runs without error and produces the assigned results • Code accounts for all possible scenario • Code is free of bugs	**10 Points - Approaching Mastery** • Completed 2 out of 3 requirements • Code runs without error • Code produces results as expected 80% or more of the time	**5 Points - Progressing** • Completed 1 out of 3 requirements • Code runs without error • Code produces results, but not necessarily the correct results	**0 Points - Emerging** • Completed none or partial out of the 3 requirements • No submission • Code runs with error

Coding Conventions/Formatting • Appropriate header, name, short description at top of the notebook • Imports are at the top of the file, just after any headers or subheads. • Files read in from relative file path • Functions and variable names are descriptive, lowercase, with words separated by underscores • Clean code, no repetition, maintainable and highly reusable code. • Appropriate code wrapping and cell sizes • Appropriate subheads as needed	**10 Points - Mastery**	**8 Points - Approaching Mastery**	**5 Points - Progressing**	**0 Points - Emerging**
Deployment/Submission • Files submitted in personal repo • Appropriate directory structure with correct files needed to run scripts • Appropriate commit messages • Appropriate README	**10 Points - Mastery**	**8 Points - Approaching Mastery**	**5 Points - Progressing**	**0 Points - Emerging**
Documentation/Comments • Code is well commented with concise, relevant comments	**10 Points - Mastery**	**8 Points - Approaching Mastery**	**5 Points - Progressing**	**0 Points - Emerging**