# Multifactor Model



## Meets Specifications

Hello there,

Congratulations on finishing the project 🎉

This was a brilliant submission. The work was exceptional! You did a great job and should be proud of yourself. After reviewing this submission, I am impressed and satisfied with the effort and understanding put in to make this project a success. All the requirements have been met successfully 💯 %

Keep doing the great work and all the best for future project.



## Statistical Risk Model

✓ **The function `fit_pca` fits the PCA model with returns.**

Well done! You correctly fit a PCA model with returns and required parameters.

✓ **The function `factor_betas` gets the factor betas from the PCA model.**

You successfully got the factor betas from your PCA model.

✓ **The function `factor_returns` gets the factor returns from the PCA model.**

You successfully got the factor betas from your PCA model.

✓ **The function `factor_cov_matrix` gets the factor covariance matrix.**

You successfully got the factor betas from your PCA model.

✓ **The function `idiosyncratic_var_matrix` gets the idiosyncratic variance matrix.**

Good, you computed the correct idiosyncratic variance matrix from the residual returns.

✓ **The function `idiosyncratic_var_vector` gets the idiosyncratic variance vector.**

Fantastic, you successfully generated the idiosyncratic variance vector as the diagonal of the idiosyncratic_var_matrix.

✓ **The function `predict_portfolio_risk` gets the predicted portfolio risk.**

## Create Alpha Factors

✓ **The function `mean_reversion_5day_sector_neutral` generates the mean reversion 5 day sector neutral factor.**

Awesome, you correctly generated the mean reversion of the 5-day sector neutral factor.

✓ **The function `mean_reversion_5day_sector_neutral_smoothed` generates the mean reversion 5 day sector neutral smoothed factor.**

Good job, your smoothed factor is successfully calculated job, your smoothed factor is successfully calculated

## Evaluate Alpha Factors

✓ **The function `sharpe_ratio` gets the sharpe ratio for each factor for the entire period.**

Fantastic, your sharpe_ratio function is correctly implemented. Sharpe ratio is the measure of risk-adjusted return of a financial portfolio. A portfolio with a higher Sharpe ratio is considered superior relative to its peers.

✓ **The student correctly mentions what would happened if you smooth the momentum factor and why.**

## Optimal Portfolio Constrained by Risk Model

✓ **The function `OptimalHoldings._get_obj` returns the correct objective function.**

Good job, you generated the correct objective function.

✓ **The function `OptimalHoldings._get_constraints` returns the correct list of constraints.**

You have the correct list of constraints. Well as regards the assertion error, it is due to conflicting packages, it has been flagged already.
I am pretty sure the content creators would fix it soon, I apologize for the inconvenience.

✓ **The function `OptimalHoldingsRegualization._get_obj` returns the correct objective function.**

Fantastic, you successfully generate the correct objective function for optimizing the returns with regularization.

✓ **The function `OptimalHoldingsStrictFactor._get_obj` returns the correct objective function.**

Getting target_weights to match the shape of the weights is key in this function, you've correctly done that.
The objective function is correctly implemented as the L2 norm of the difference between weights and the target_weights.