

[← Back to AI for Trading](#)

Breakout Strategy

REVIEW

CODE REVIEW

HISTORY

Meets Specifications

Bright Learner,

My congratulations for this magnificent submission. You have passed this project successfully and addressed all the rubrics. You can be proud of yourself. Thank you and continue to work hard and make efforts in your submissions.

I wouldn't want to take much of your time but, can you please do well rate this review from the star ratings and provide some feedback on the `review_rating_comment_section` on how long it took you to get this awesome work done and the biggest challenge you faced in finishing this project. Getting your feedback is always a pleasure, I will be very gladly to hear about your thoughts :) Thanks in advance!

Generate Signal



The function `get_high_lows_lookback` computes the maximum and minimum of the closing prices over a window of days.



The function `get_long_short` computes long and short signals using a breakout strategy.



The function `filter_signals` filters out repeated long or short signals.



The function `get_lookahead_prices` gets the close price days ahead in time.



The function `get_return_lookahead` generates the log price return between the closing price and the lookahead price.



The function `get_signal_return` generates the signal returns.

Evaluate Signal



Correctly answers the question "What do the histograms tell you about the signal returns?"

Thanks for meeting the specification in this section. You greatly provided a reasonable discussion regarding the signal return. Great answer!

Outliers



The function `calculate_ks_test` calculates the ks and p values.



The function `find_outliers` returns the list of outlying symbols.

This implementation is impeccable. 🙌 The required changes made by the previous reviewer has been addressed perfectly.

```
""" Symbols that are outliers """
#TODO: Implement function
ks_value_outliers = ks_values > ks_threshold
p_value_outliers = p_values < pvalue_threshold
ks_and_p_value_outliers = (ks_value_outliers & p_value_outliers).where(lambda x: x == True).dropna()
outliers = set(ks_and_p_value_outliers.index.values)

return outliers

project_tests.test_find_outliers(find_outliers)
```

 [DOWNLOAD PROJECT](#)

[RETURN TO PATH](#)