Accepting the Unexpectable

Duncan Macgregor and Nat Pryce, SPA2014, Wednesday 2nd July, 2014 2pm-3:15pm

Introduction

Nat's experience of building a financial analysis software, using a library developed by another team who were using TDD. The library was supposed to make projections of future interest rates over time based on historical data. All Nat's tests failed when taking a new release.

Plotting the numbers produced showed that the graph showed a clear "kink" in the projection graph. Yet it passed the unit tests – how come?

Anscombe's Quartet demonstrates that mean, standard deviation, linear regression and variance of a line plotted through four sets of points can be identical, even when the plots of the underlying data sets are self-evidently very different.

So visualisation is very very important for picking up macro effects. Yet FIT tests etc. only present tabular data.

What is Approval Testing?

Interesting things about interest rate forecasts:

- There is no objectively "correct" answer
- Noisy data
- Element of "expert judgement" when it comes to acceptability

Llewellyn Falco promotes Approval Testing as a technique to complement Specification by Example.

Specification By Example vs. Approval Testing

SBE takes a set of example data to create a set of exemplars in a table (typically) that specifies significant test cases. You then have to write test fixtures to inject those values into the system under test and compare the outputs one at a time with the expected results.

Approval testing, on the other hand, just runs a test and compares the result with a "golden copy" of the expected results. Any differences are examined and categorised into "test failure" or "specification change". The new output is thus selectively merged with the previous golden copy to form a new golden copy.

The results can be projected through various visualisations to help interpret them and decide whether the previous or new output is "better". No need to parse the data! This saves a huge amount of data.

Exercise

Clone the git project and bring it up into your IDE.

Get the tests to pass. See the effect on the HTML output of changing degree of curve-fit, reading the data column containing seasonally adjusted ppm CO_2, and so on.

Conclusions

HTML was an easy way to visualise the data in this environment, but not necessarily ideal in a commercial situation.

Incorporate visualisations in acceptance test reports.

Identify the data that is stable (and can be used as values in tests) and that which is unimportant / volatile and should only be included in visualisations for approval by domain experts.

Difference between approval and acceptance? Approval = review differences between expected and unexpected results. Acceptance = evidence-based sign-off (some of which can be automated).

Resources

http://github.com/dmcg/acceptance-and-approval-workshop