For Benedict

Last modified: Friday, July 18, 2025, 12:20

Ideas (at least partly) implemented:

1. ROC for normality test (classification methods). We have also discussed the effect of sample size on this, but haven't fully explored that yet.
   We can think of a p-value threshold which is a function of n. Or use something like a D-value instead of p-value.
2. ROC-like curve for downstream test - Type-I error control for the normality test is useless anyway. Use different thresholds of the normality test to obtain EPG and EPL of downstream. To add the Bayesian idea of prior probabilities to the axes. Soon to do: Actual ROC for downstream and then add the AUC for varying sample size.
3. Older version of AUC across varying sample sizes etc. These need to be updated.

Ideas to implement soon:

Chapter 1 (Utility of normality tests):

1. Clearly write the specific aim for the utility project - There is cost for doing a normality test, but there may be some benefits too - which one is more in a particular situation? Can we quantify the cost and benefit? We aim to provide a specific answer through cost-benefit analysis.
2. Create 2-3 examples (for example a two sample t-test, a one sample t-test, and a regression) to demonstrate different pieces of the whole framework. Have a number of different test options for each case.
3. One of the two user-provided tests can be - do a transformation (e.g. Box-Cox) and run the parametric test. We need to implement this option.

4. We will do simulation studies to demonstrate our framework for a number of situations with varying sample sizes, data generating distributions (perhaps include some discrete distributions too), normality tests, downstream tests etc. Based on this, we will provide some suggestions for some commonly used testing situations. Suggestions can be about when to run normality tests, what normality test to use, what effect size measure to use if we advocate effect sizes, etc.
5. We are suggesting measures of utility for known non-normal distributions. But the user may just provide a pilot data. We can do a simulation study to compare the true utility vs estimated utility.
6. We should keep in mind computation costs for tests such as permutation test. The user interface should also output the computation time for each of the test options.
   By the way, it is also unclear what test statistic one should use with the permutation test. The test options provided by the user could even be permutation test with two different statistics -e.g. mean and median.
7. For the ROC-like curve, the user could have a Bayesian approach: Prior probability of normality vs non-normality. This can later be a nice simulation study framework where the probability of non-normal can be treated as the effect size.
8. Suppose, F, the data generating cdf, belongs to {F0, F1}. Estimating which one it is. Given correct estimation, the conclusion is obvious, the issue is the estimation inaccuracies. Think Bayesian in this context.
9. Some discussion of the classical large sample results (e.g. Fisher Behren's problem, relative efficiency of Wilcoxon vs normal) need to be included in the writeup.
10. On this note, we should talk about some large sample results for our one methods, e.g. consistency of our Monte Carlo type methods.
    Also, rejection criteria are based on quantiles of the distribution of a statistic. If the statistic converges to normal, e.g., then how does the quantile converge? Of course it will converge, at least for continuous distributions, based on the definition of convergence in distribution, but how fast etc? That will give us an idea about the asymptotics of the errors.
11. Think about and discuss some application areas where this matters. For example, discuss importance of distributional assumptions for commonly

used tests in genomics.

Chapter 2 (Machine learning method):

1. Fuller exploration of different normality tests. This is not immediately needed, but we may want to include some more recent developments in this field - for example some combination tests. We also want some visual methods implemented, e.g. something like (a better version of) the one based on QQ-plot that you used in STAT 6203 project.
2. Then we can compare our machine learning method with the top candidates. Larger simulations can be done to show its effect on downstream tests just like we did simulations using existing normality tests in chapter 1.
3. Gain better understanding about the theory behind the different learning methods. This will tell us what a certain method actually does and why something might work better than something else. A literature review of these will be needed for writeup.
4. Develop some ideas about how we can use the variable importance scores.
5. This is probably a long shot, and perhaps will just be listed as future research directions - Can we classify as Problematic vs not-problematic as opposed to normal vs non-normal.

Chapter 3 (Selective inference issue for normality tests):

1. Create at least one clear example of the danger of selective inference.
   (a) One example where the conditional type-I error is highly inflated leading to high overall error.
   (b) Another example where the use uses a transformation - which is a conditional-type approach. Demonstrate that fishing for transformations can be dangerous.
2. Implement the sample splitting approach. Explore how well it does to resolve the selective inference issue.
3. Read more to come up with more ideas. We need at least one more new idea for this chapter. We need to show its promises before the qualifying exam. We can then do a fuller exploration after the exam.

4. Discuss a little bit about possible multiple testing approaches to help with selective inference.
5. Show that the problem of selective inference is likely to be less severe if the normality test is really good. Hopefully we can then justify that the machine learning approach will be helpful alleviating some of the selective inference concerns too. In this context, think about the role of sample size. Some large sample results will be useful.