Kernel Density Estimation And Hypothesis Testing on Persistence Diagrams

Mike Wu

September 29, 2015

1 Kernel Density Exploration

Strangely enough, the axis on the TDA website are flipped when compared to the ones generated using the same code provided by the tutorial. Most likely, this is an outdated tutorial. Or perhaps, I must manually flip the axis? More evidence of a possible typo: The rotated and bar plots look rather identical to the plots in the tutorial so it should be doing the same thing. Given that, why does the 1st order feature appear first? And why do so many additional 0th order features appear far later? Are these noisy perturbations?

To peek deeper into what KDE does, let's try to plot the 3D topology of the smoothed gaussian curves. First thing I notice is that things are extremely smoothed and that there exists only 1 obvious local maxima. If that is true than the KDE diagram makes sense since a 0th homology feature is found first (the tall peak) and then the 1st homology feature is found (the rest of the smooth circle). The residual 0th homologies, like the bands describe, are due to small bumps on the surface, i.e. noise perturbations. (Does this make sense?).
Observation 0: The larger the object, i.e. radius of larger size, the smaller the gaussian peaks overlap, an the less prominent the overlaps. A circle of radius 10 will probably have many more small short-lived 0th homology features and a less prominent 1st homology feature than a circle of radius 1 (But when I ran it, the two actually look very similar, perhaps we can discuss it more).

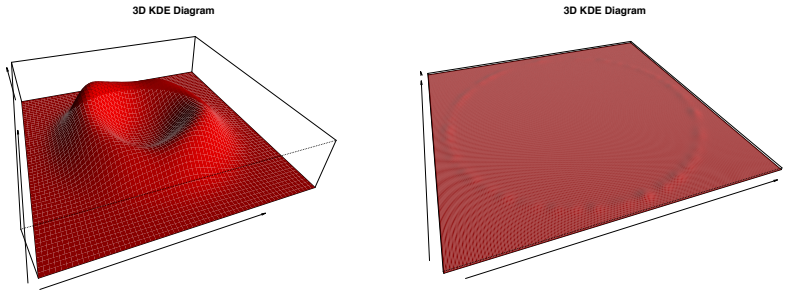

Figure 1: 3D graph of KDE for a uniform circle of radius (left) 1, (right) 10.

The algorithm has been reformatted to fit the TDA guide as well as add an

additional confidence band (new Github commit). The objects above the band indicate those are statistically differentiable from noise.

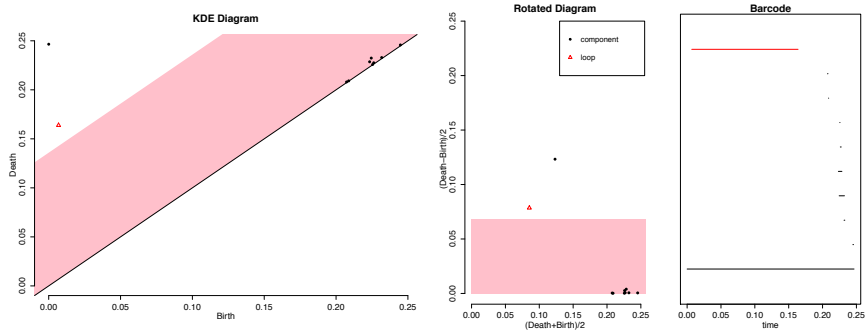

Figure 2: (Left) Regular KDE Diagram from a uniform circle with a band representing a significance level. (Middle) Rotated KDE Diagram. (Right) Bar Diagram.

We can also explore the effect of different parameters on the KDE system. **Observation 1:** the more samples generated from each uniform circle, the smaller the area imputed to noise. This makes sense since we are more confident that a circle exists, and attribute less of the space to random objects. **Observation 2:** h is the smoothing factor, and the higher the smoothing factor, the more confident we are that less area is attributed to noise. This also makes sense, since increasing h is almost equivalent to applying a stronger gaussian smoothing filter, whose primary role is to nullify noise. Notice in this case we didn't lose much resolution because there is such a clear defined object. In more difficult cases, increasing h too much probably loses resolution. **Observation 3:** The grid for which KDE is built on affects the confidence level on noise. Very small grid sizes have lower confidence intervals, and as you increase the grid sizes the noisy areas increase, probably suggesting that we are losing resolution. Strangely enough as you increase the grid even more, to remarkably high levels, the noisy areas diminish fast with seemingly no loss in homological objects. I am not sure why this is happening?

2 Noisy Kernel Densities

Last week, I investigated the Rips Complex and noticed that it does not handle noisy data very well. Because it gives each point equal weight, noise easily clouds any true structure. Intuitively, kernel density should be more robust since noise would be equivalent to small maxima.

2.1 Perturbing Points on the Circle

Each of the K points sampled from the uniform circle is shifted a little by a small Gaussian noise. Let's go with something extreme and sample from $\sim N(0,1)$ with a circle of radius 5. Generally, it seems like this definitely does handle noise better, but adding some noise does produce a lot of short-lived 1st homology

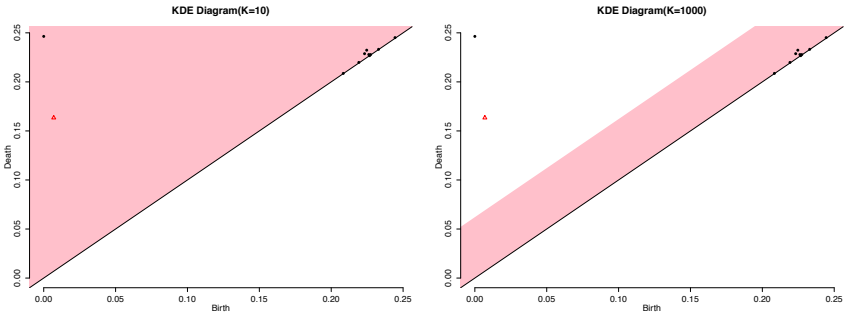

Figure 3: (Left) KDE of uniform circle with 10 points. (Right) KDE of uniform circle with 1000 points.

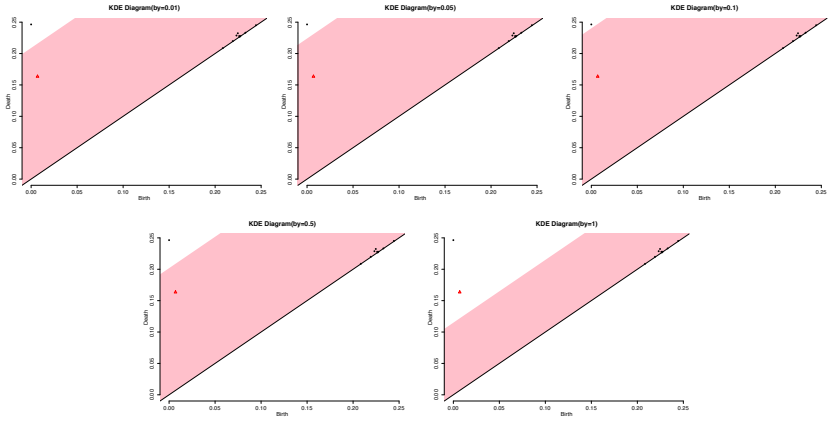

Figure 4: Persistence diagrams of a uniform circle with different grid step sizes. The amount of noise coverage is seemingly parabolic with step size.

features and long-lived 0th homology features, which may distract from the true structure. This may have important consequences when conducting hypothesis tests between persistence diagrams since as figure 5 shows, the persistence diagram of noisy data is also noisy.

2.2 Adding Additional Points

Instead of perturbing the points themselves, additional gaussian samples are added around the circle. This should again, be an easier problem than the previous one. There still is a clear circular structure, albeit distractions all around. Looking at the data, the main loop was not found. All of the 1st order homology features had very short lives, indicating that could not have been the loop. And if they were, they must of immediately been distracted by the noise. From these observations, I think that Kernel Density Estimate also does not do well with noise. In general, I think topological methods are not good at handling noise. Intuitively it is hard to not consider a point if you do not know

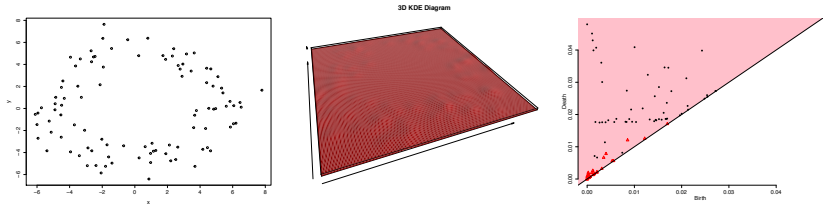

Figure 5: (left) Noisy circle data. (middle) The topology of the perturbed circle. Notice the several clumps of local maxima. (right), A single persistence diagram. There are several shortlived 1st homology features, and only a couple persistent ones. There are several long-living 0th order homology features.

the final object's shape. Then, it's possible to find objects that are not actually there. **The only solution is to rerun the simulations over and over and look for significant objects.** But even then, I am not sure if the object that appears is noise or significant. Actually... after reading the paper, it seems that KDE is probably really good at handling noise but just not as good as Rips in discovering topological features. So although we can't find the loop, that isn't actually because of the noise, but because of the method itself!

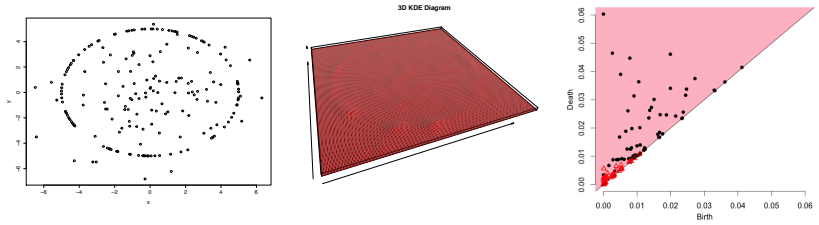

Figure 6: (left) Noisy circle data. (middle) The topology of the circle w/ extra points. Notice the several clumps of local maxima, esp in the middle. (right), A single persistence diagram. There are several shortlived 1st homology features, and several long-living 0th order homology features.

3 Minimally Noisy Kernel Densities

Another possibility is that I just added too much noise. Perhaps KDE is resistant to noise but I perturbed it to the degree in which even a hypothetically optimal topological algorithm could not distinguish them. Repeat the process with a reduced noise setting:

Consider a dataset of a uniform circle of radius 1. The persistence diagram (figure 7) shows a very clear and long-living 1st order homology feature. This is a very clear dataset. Continuing through figure 7, both forms of noise addition shifted the persistence diagrams slightly, but not to a drastic manner. All persistence diagrams still show the clear loop. This suggests that in situations of light noise, KDE is a much better option than Rips. However, if the Universe

simulation data is anything of what I expect, the data will be much messier. How will KDE be able to hold together then?

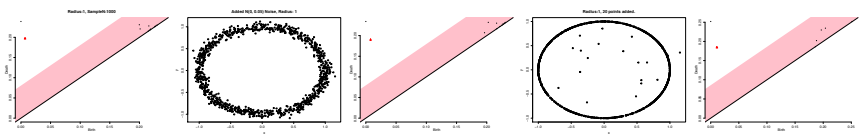

Figure 7: (1) Original plot (no noise added). (2,3) Plots for point perturbation. (4,5) Plots for point addition. Each contains plot of circle and a persistence diagram.

4 Persistence Diagrams on Hypothesis Testing

To properly compare persistence diagrams, a version of hypothesis testing adapted from Robinson, Turner 2013 (http://arxiv.org/pdf/1310.7467v1.pdf) of randomization-style null hypothesis significance tests (NHST) is used. Instead of the bijection distance advocated by Robinson and Turner, the 2-Wasserstein and bottleneck distances are used. (All code for this is in the Github). The implementation of random sampling with NHST remains faithful. As predicted, the expectation of Z (our estimation for p-value) approaches the true p-value as N grows large.

To valid correctness, the same simulated example is included as in the paper. There are two shapes, the first being a uniform circle of radius 1, and the second being two uniform seconds joined by 1 point, 1 of radius 3/5, one of radius 2/5. Five simulations are conducted with each simulation holding 20 point masses for each of the 2 circles. Persistence diagrams are evaluated on the two point masses, and inputted into NHST. Gaussian noise is added to the circles for each of 20 point masses. Different simulations are conducted with different levels of gaussian noise to see how the test handles perturbations.

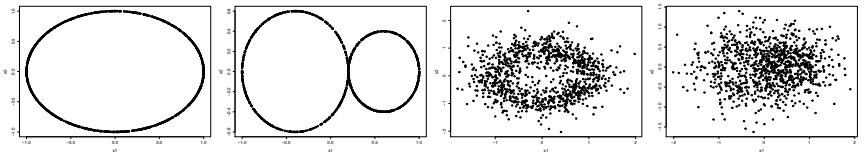

Figure 8: Left 2: Simulation comparison data set (0 noise). Right 2: Same data set with Gaussian Noise (std=0.1).

Observation 1: The testing is rather resistant to noise! Notice in the figures above how much noise 0.1 standard deviation already adds. However, with NHST, a noise level of 0.25 std only creates a p-value of 0.17. (Of course, the baseline test shows that a noise level of 0 produces a p-value of 0 – as in there is a probability of 0 that the two are the same. Concurrently, a high noise level of 1 produces p-values of 0.8-0.9, suggesting that they are the same).

Observation 2: As noise increases, the confidence interval for p-values increase dramatically. With a variance of 1, the noise has a very wide spread. Because of that, depending on the sampled noise, the two objects

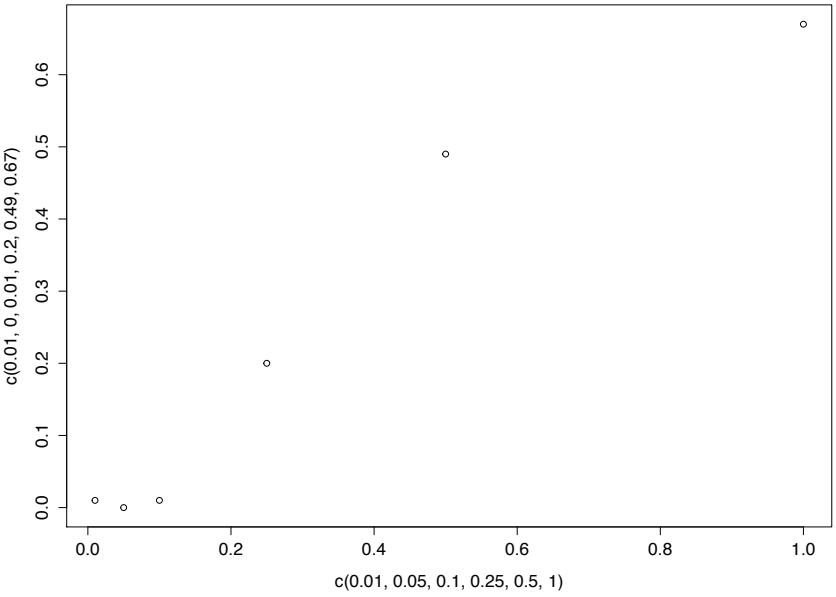

Figure 9: The p-values for different levels of variance (x-axis).

could be really close (really noise) or really different (really clear). In general, the fact these have such large variances with noise might be a problem.

Observation 3: As N increases, the p-value stabilizes. This is just the law of large numbers, so it is not anything we should be surprised by.

End Note: I think this is ready for use!