Eigenvalue collisions for approximations of certain R-diagonal elements

Carlos Vargas

January 1, 2025

Abstract

We study the collisions of eigenvalues for a family of matrices

$$R(s,t) = \alpha(s)C + \beta(s)U(t),$$

periodic on the parameter $t \in [0,1]$, with $s \in [0,1]$, determined by a realization of a Ginibre matrix C and a rotating unitary matrix $U(t)$.

The matrices in this family are approximations of R-diagonal elements in free probability, which have been studied by Haagerup-Larsen [?], Kemp-Speicher [?] and P. Zhong [?].

We study the eigenvalue collisions for all values of s and t. To do this, we fix s, and increase the periodic value t, while keeping track of each eigenvalue.

Unless s is too close to 0 or 1, the process leads in general to a non-tivial permutation $\sigma(s)$, with pleasing visualizations of flowing, repellent eigenvalues.

The intricate web of paths that the eigenvalues collectively traverse remains quite stable for small variations in s. However, the actual permutations $\sigma(s)$, $\sigma(s+\Delta s)$ do present variations, indicating eigenvalue collisions at some intermediate values (s,t), $s \in (s, s+\Delta s)$, which explain the permutation discrepancy and are indicative of local maxima of the speed of eigenvalues.

The eigenvalue 'track-flips' that occur before/after these colissions account for the essential differences between consecutive eigenvalue trajectories, with the eigenvalues proceeding to team-up to whirl, first around themselves and then around zero, as s increases.

We report some first statistics about these processes and their collisions, and we include a simple package to perform/store/display these (parallellizable) computations and visualizations.

1 Introduction

Let C be a realization of an $N \times N$ Ginibre matrix and let ω be the first clockwise non-trivial complex N-th root of 1. For $t \in [0,1]$ let

$$U(t) = \mathrm{diag}(\omega^{tN}, \omega^{tN+1}, \omega^{tN+2}, \ldots, \omega^{tN+N-1}),$$

be a diagonal matrix with equi-distant points along the circle. Notice that $U(0) = U(1)$.

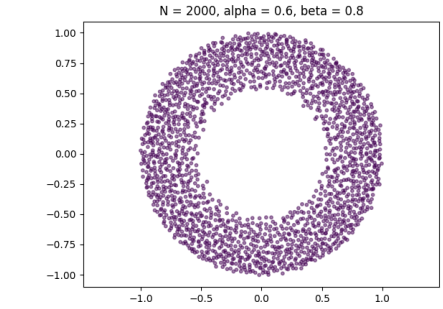
We want to study the eigenvalues of the matrix model:

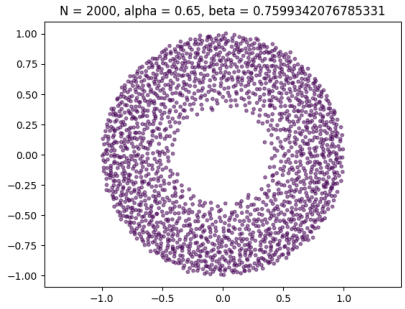
$$R(s,t) = \alpha(s)C + \beta(s)U(t), \quad \alpha(s) = \cos((s\pi)/2), \beta(s) = \sin((s\pi)/2)U(t),$$

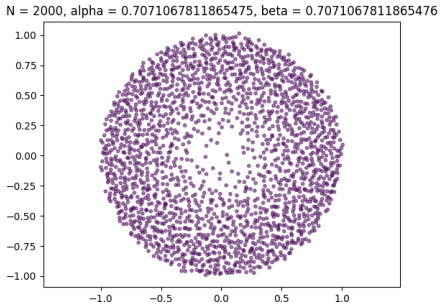
which is periodic on $t \in [0,1]$ for fixed s.

In general it is quite a task to compute distributions of non-selfadjoint random matrix models. The matrices in this model, however, are approximations of R-diagonal elements in free probability, for which some actual computations are possible [?,?].

Depending on the values of α and β, the asymptotic distributions are supported on a centered annulus or disk, (see Figure).



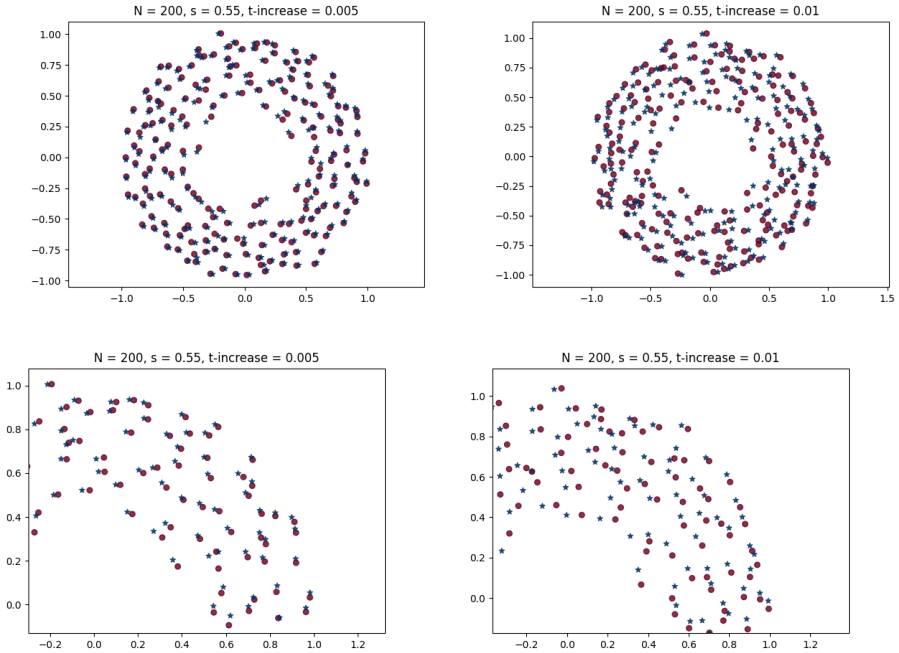




We choose $\alpha(s) = \cos(s\pi/2)$ and $\beta(s) = \sin(s\pi/2)$ specificaly to make the outer radious of the model equal to one. The eigenvalues for any other positive pairs (α, β) can be obtained by scaling values in this parametrization.

In this work we mainly want to draw attention at the effect of increasing the periodic parameter t, which 'turns' the eigenvalues of the model clockwise.

The figures show one set of eigenvalues (red circles) trailing the next set of eigenvalues after small increases of t (blue stars)



Notice that the eigenvalues in the outer part of the domain are moving much more slowly than the eigenvalues on the inner part.

On the other hand, since $U(0) = U(1)$, as t increases from zero to one, most eigenvalues won't make a complete turn to return to their original positions, but each position still must be reached at $t = 1$. This leads to a non-trivial permutation $\sigma(s)$ associated with the matrix process.

We use this permutation to detect and count eigenvalue collissions (see Section 2).

The rotating unitary matrix can actually be replaced by any other parametrized curve(s).

(Include circuit and crossing images, comments about increase in number of collisions for circuit vs circle, 'cheating' Madrazo eigenvalues in crossing case)

We include implementations of our methods in Python to reproduce the data here presented and for the readers convenience.

2 Permutation $\sigma(s)$

Unless s is too close to 0 (static) or 1 (perfect rotation) the process of increasing the periodic parameter t rotates inner and outer eigenvalues at different speeds, leading to non-trivial permutations $\sigma(s)$ relating the eigenvalues of $R(s,0)$ and $R(s,1)$, after increasing t from zero to one.

We want to study eigenvalue collisions aided by this permutation.

Let us consider first the eigenvalues of $R(0,t) = R(0,0)$. These are just the eigenvalues of the complex Gaussian Ginibre matrix C, with explicit joint distribution

$$\rho(\lambda_1, \ldots, \lambda_N) = \frac{1}{\pi^n \prod_{k=1}^{N} k!} \exp(-\sum_{k=1}^{N} |\lambda_k|^{-2}) \prod_{1 \le j < k \le N} |\lambda_k - \lambda_j|^2$$

As $N \to \infty$ this distribution converges to uniform distribution on the unit disc.

To be able to store the relevant data properly, we will first label the eigenvalues increasingly by norm at $R(0,0)$.

Then we consider all the values $R(s,0)$, increasing s slowly, keeping track of the eigenvalue labels continuosly, until we reach $R(1,0)$.

In case of ambiguity of eigenvalue tracking, a refinement is performend on the partition on $s \in [0,1]$.

After this step a continous ordering of the eigenvalues for all values of s and $t = 0$ has been achieved. Now we consider, for each fixed $s \in [0,1]$ (in the s-partition) the process $R(s,t)$ when increasing t from 0 to 1, keeping track of the eigenvalues.

For small variations of s we may observe differences on the permutations, while noticing very little discrepancies on the collective trails traveled.

In the Figure $s = 0.0500, 0.0505, 0.0510, 0.0515$. The eigenvalue tracks are colored according to cycle length, from yellow (shortest cycles) to purple (longest cycles)

Notice first the large, 26-element purple cycle $[1, 7, 19, 51, 25, \ldots, 16, 8]$.

The collective paths remain almost unaltered, but there are some few eigenvalue collisions in between each frame:

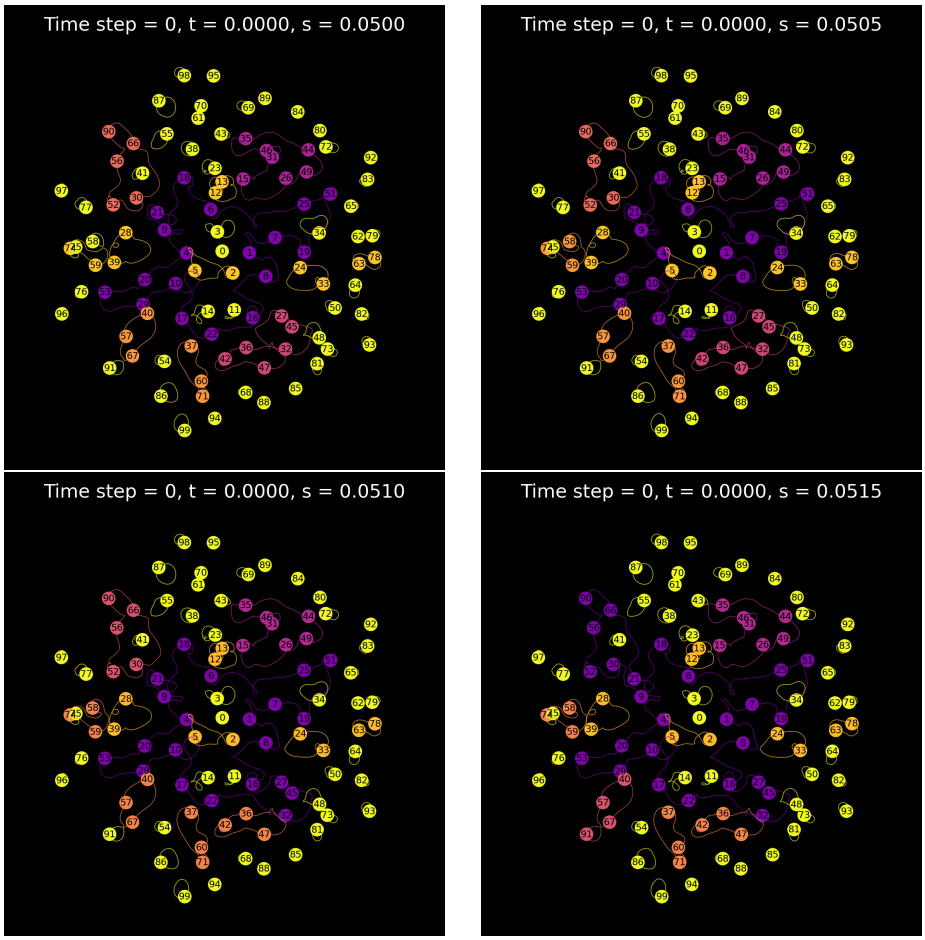
The corresponding permutations $\sigma(s_0)$, $\sigma(s_1)$, $\sigma(s_2)$, $\sigma(s_3)$ all present small differences. These are witness to *eigenvalue collisions* at some values (s,t), $s \in (s_0, s_3)$.

The collissions explain the permutation discrepancies in the following way: if the eigenvalues i and j collided, and they were originally pointing to k and l, they will point after the collision to l and k instead.

The eigenvalues keep their initial tracks, but switch paths after the value of t where the collision occurred.

In the figure, there is a collision of the singleton [58] and 59, from the cycle $[59, 74]$ between s_0 and s_1 that makes produces the cycle [58, 59, 74] in the next frames.

The next permutation configuration $\sigma(s_2)$ is explained by two collisions:



One collision (27 vs 36) splits the 6-cycle $[32, 45, 27, 47, 42, 36]$ into two cycles $[32, 45, 27], [47, 42, 36]$.

The second collision 27 vs 22 (or the first, we are not sure of the order of the collisions at the current refinement) joins the cycle $[47, 42, 36]$ with the large purple cycle.

Finally, the last permutation is explained by two collisions: One involving 91 and 67 which joins the singleton $[91]$ from the cycle $[67, 57, 40]$, to produce the 4-cycle $[67, 91, 57, 40]$

The second is crash involving 21 and 30 which results in incorporating the 5-cycle ok 30 to the big purple cycle.

Thus, we use these permutation discrepancies to detect eigenvalue collisions. From computations for small N we conjecture that there are exactly $N(N+1)$ collisions as s goes from 0 to 1.

We report some first statistics about these processes and their collisions.

There is an interesting aspect about counting collisions: if a collision occurs between two eigenvalues which are both not currently singletons, then it is ambiguous to simply report that a given pair of eigenvalues crashed.

The pairs of eigenvalues that acually crash depends on how we reach $R(s_0, t_0)$. Did we go in a straight line from $(0, 0)$ to $(s_0, 0)$ and then rotated?, or were we allowed to rotate at intermediate values s, $0 < s < s_0$?

We include a simple package for these computations and visualizations.

From the t-processes we may extract different types of eigenvalue collision data for visualtization and/or statistics.

3 Collision statistics

Diagonal matrices vs free case.

4 About the algorithm

General description of the algorithm.

Main bottleneck: linear algebraic library method for eigenvalues. Maybe a more specialized, explicitly pivoted method could reduce computing time at this step.

Use of Delaunay Triangulations.

About condition number and shift.

5 References

- P. Zhong. (include reference...)