# Contoured Attribution Map: Explainable AI (XAI) for Human-in-the-Loop Hard Real-Time Computer Vision Systems

**Ryan Ceresani**
Johns Hopkins University
Whiting School of Engineering for Professionals

## Abstract

Systems with hard real-time requirements often have high consequences for failure. Current forms of network attribution are not designed for real-time human-in-the-loop systems. This environment does not require a mathematically comprehensive solution, but rather a human readable one. By extending the existing attributions heat map algorithms with quick human perception enhancements, a modified **contoured attribution map** provides a non-AI operator confidence in understanding what the network used to make decisions.

## Explainable Computer Vision

With the increase in applications using computer vision and convolutional neural networks (CNNs) came an increase in the desire for creators and users to better understand **why** a deep network made a certain decision. Earlier research into this focused on using input activation (Erhan et al. 2009), explanation vectors (Baehrens et al. 2010), and saliency maps (Simonyan, Vedaldi, and Zisserman 2014). The research in this area is on-going with new methodology constantly being developed for primary, layer, and neuron attributions. This research is faced with the challenge of evaluating the efficacy of the algorithms empirically. Image detection and classification can rely on human labeling to provide comparisons, but network attribution must rely on subjective analysis and mathematical examination.

This paper looks to identify a method for presenting one of the newer attribution algorithms in a method that is easier for humans to parse. It is beyond the scope of this research to examine the underlying algorithm for validity.

### XAI on The Edge

One unifying factor for these algorithms is a "density" with which they are approached. The focus of the cited papers (above and below) is a mathematical precision and inscrutable scientific application. Visualizations produced may be illuminating for AI researchers, but do not cater to a lay user.

There is a subsection of computer vision applications which require an operator or human-in-the-loop (HITL) for the decision making process. A 2018 study by Stanford concluded that the HITL system performed better than either human or AI in isolation (Bien et al. 2018).

The need for explainability grows as the consequences of decisions increase. Consider the following scenario: A remote controlled military drone is escorting an allied convoy, providing close air support. An on-board AI system identifies what it believes to be a hostile with a shoulder-launched missile platform. There is limited time to react to the situation. This poses an ethical question outside the scope of this paper. However, practically, an operator would want to know the network was influenced by the "right" thing before authorizing a strike. This example highlights the need for easy to understand network attributions.

### Integrated Gradients and SmoothGrad

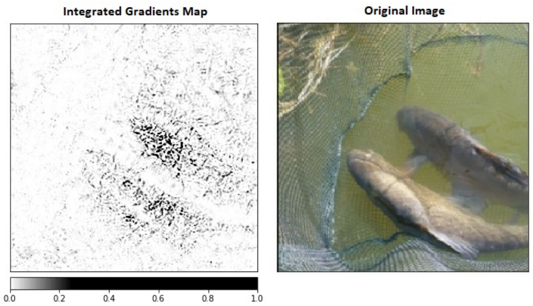

Figure 1: Integrated Gradients Attribution Map of two fish (original image from ImageNet (Deng et al. 2009))

The integrated gradients method (Sundararajan, Taly, and Yan 2017) of network attribution, offers an *axiomatic approach* (2017) to the empirical evaluation issue. A key component of this is the identification of each individual pixel's influence. The prescient part of this is the ability to produce a heat map from the results. [see Fig. 1]

One noticeable issue with the resulting heat map from integrated gradients is a relative lack of "sharpness" in the image. There are clearly defined sections, but large amounts of noise. The introduction of *SmoothGrad* (Smilkov et al. 2017) proposed a useful method for refining the visual characteristics of what they term sensitivity maps. (Analogous with the previously termed heat maps in this paper.) Smilkov et al. acknowledged that previous work on gradient or back-propagation solutions attempted to address the key issue of over-saturation, but still contained rather noisy sensitivity maps. The *SmoothGrad* method involves not using the gradients directly, but rather a smoothed gradient using a Gaussian Kernel. (2017) The paper subtitle explains it as "removing noise by adding noise."

Figure 2 demonstrates the result of passing Figure 1 through *SmoothGrad*.

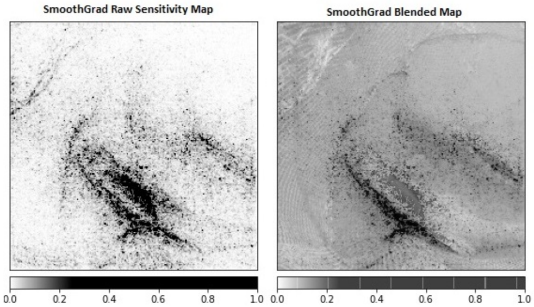

Figure 2: SmoothGrad Raw and Blended Maps of two fish (original image from ImageNet (Deng et al. 2009))

The results sharpen the resulting regions and make the important pixels more visible. The specific regions of interest are quickly identifiable but for a HITL system, it is not as defined for relating it back to the original image. Additionally, the mapping of individual pixels still provides a minor level of noise and distraction. The proposed solution to this is a *Contoured Attribution Map*.

## Contoured Attribution Map

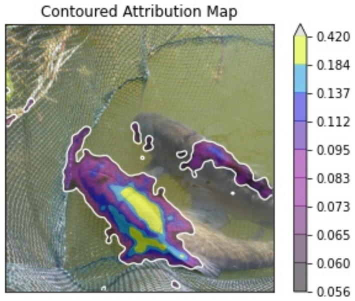

Figure 3: Contoured Attribution Map (original image from ImageNet (Deng et al. 2009))

To create a contoured attribution map, the sensitivity map of *SmoothGrad* is passed through a two dimensional Gaussian filter:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

The expected characteristics of the pre-contour attribution maps should determine the size of the Gaussian kernel. Figure 3 uses a **13x13** kernel. The top $15\%$ of the resulting blurred attribution map is quantized into 10 sections and their contour is applied over the original image.

The above operations intend to specifically draw attention to the highest influential areas and allow for split second "go-no-go" decision making. The underlying results of attribution do not drastically change from the *SmoothGrad* results.

## Attribution Map Study

The intended target of *Contoured Attribution Maps* is human interaction and the efficacy of the maps for that purpose can only be validated through a study. A reaction time test was created to measure the participant's reaction time when presented with one of three styled attribution maps.

### Study Setup

Participants were presented a series of 20 images. The image was either Raw *SmoothGrad*, Blended *SmoothGrad*, or *Contoured Attribution*. There were three possible sets of images to be presented, each with a different version for each image, so that a participant did not see the same image in multiple forms. The prompt for participants was as follows:

> As **quickly** and **accurately** as you can, click on the general pixel location that you feel confident is the "**most**" important location.

The reaction times for each image were recorded and, at the end, participants were asked an opinion of which style they felt they preferred.

### Result Analysis

The study had 34 participants take the timing test. The raw results are located in the appendix. The worst $5\%$ of individual times were removed as outliers or errors. This left a total of 646 remaining timing samples. The statistical results for the raw data can be seen in Table 1.

Table 1: Raw Timing Test Results

| Map Type | # Obs | Mean | Variance | Skew | Kurtosis | Min | Max |
|---|---|---|---|---|---|---|---|
| Contoured | 226 | 2.404 | 1.749 | 1.198 | 0.83 | 0.363 | 6.457 |
| Blended | 211 | 2.343 | 1.689 | 1.426 | 2.062 | 0.356 | 6.955 |
| Raw | 208 | 2.529 | 2.293 | 1.609 | 2.634 | 0.351 | 8.336 |
| All | 646 | 2.428 | 1.906 | 1.421 | 1.884 | 0.351 | 7.83 |

The mean values for each type of map are within $200ms$ of each other. The average reaction time to visual stimulus is roughly $250ms$. (Jain et al. 2015) This does not account for any actual comprehension or understanding of the image, so the raw results indicate a negligible difference in methods. This is better seen in the density distribution plot from Figure 4.

### Result Bootstrapping

The results have a positive skew, but display what could be a roughly normal distribution. Because our sample size is small we can leverage bootstrapping (Efron 1992) to expand the population based on the available scores.

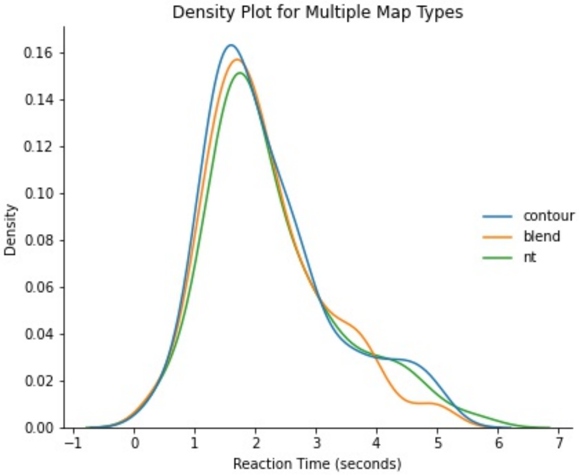

Figure 4: Density Plot of Results

Table 2 shows the statistical results from the data bootstrapped to 10,000 observations. The difference is really only visible in the density plot Figure 5, which would visually indicates that *Blended* had a slight speed advantage. However, as we can see, the differences are not significant.

Table 2: Bootstrapped Timing Test Results

| Map Type | # Obs | Mean | Variance | Skew | Kurtosis | Min | Max |
|---|---|---|---|---|---|---|---|
| Contoured | 10000 | 2.405 | 0.008 | 0.067 | -0.067 | 2.123 | 2.81 |
| Blended | 10000 | 2.342 | 0.008 | 0.05 | 0.015 | 1.959 | 2.712 |
| Raw | 10000 | 2.529 | 0.011 | 0.127 | -0.022 | 2.145 | 2.96 |
| All | 10000 | 2.428 | 0.003 | 0.066 | -0.013 | 2.216 | 2.641 |

The time differences across each map type for the whole group were not effectively different. This could indicate two things.

- The methods are equally understandable.
- The majority of images were easy to understand, regardless of methodology.

A key discriminator in the methods would be how much it mitigated slow-responses.

## Slowest Respondents

The results from the slowest 25% of participants, based on average response time, were calculated in the same fashion as above. Table 3 shows that there is a slightly larger relative decrease in time for the mean and minimum when using *Contoured Attribution Maps*. More noticeably, the variance and maximum value were drastically reduced.

This indicates that *Contoured* maps avert confusion on the harder images for those users who are already slower to react than normal. The group of participants was of a varied background with some being AI practitioners and other lay people.

We can also apply the bootstrapping methodology to the same set of results. The density plot [Figure 7] demonstrates

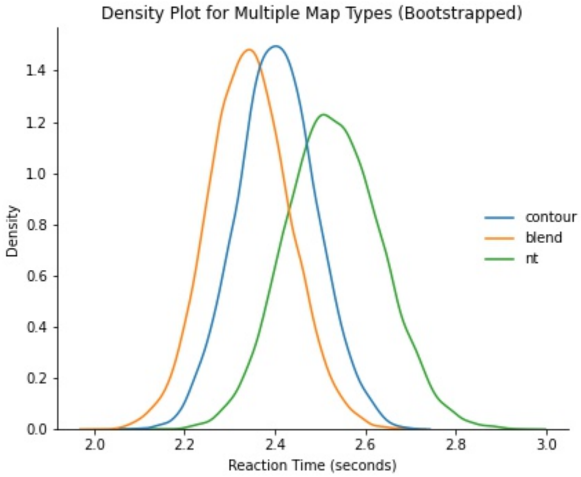

Figure 5: Density Plot of Bootstrapped Timing Test Results

Table 3: Timing Test Results (Slowest 25% Participants)

| Map Type | # Obs | Mean | Variance | Skew | Kurtosis | Min | Max |
|---|---|---|---|---|---|---|---|
| Contoured | 59 | 3.858 | 3.389 | 1.055 | 0.607 | 1.086 | 8.948 |
| Blended | 55 | 4.126 | 7.358 | 1.199 | 0.457 | 1.24 | 11.476 |
| Raw | 56 | 4.054 | 6.382 | 1.098 | 0.382 | 1.321 | 11.278 |
| All | 171 | 4.044 | 5.772 | 1.2 | 0.75 | 1.086 | 11.476 |

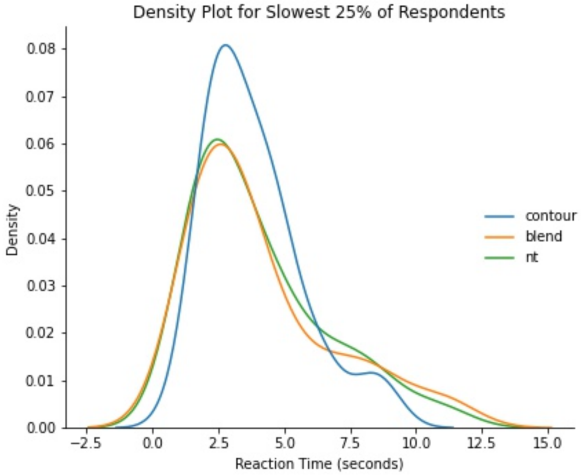

Figure 6: Density Plot of Results for slowest 25% of participants

that with a much steeper curve for the *contoured* map timings in .

Table 4: Timing Test Results (Slowest 25% Participants)

| Map Type | # Obs | Mean | Variance | Skew | Kurtosis | Min | Max |
|---|---|---|---|---|---|---|---|
| Contoured | 10000 | 3.861 | 0.057 | 0.107 | 0.045 | 2.985 | 4.713 |
| Blended | 10000 | 4.119 | 0.13 | 0.185 | -0.004 | 2.877 | 5.842 |
| Raw | 10000 | 4.057 | 0.111 | 0.111 | -0.006 | 2.858 | 5.638 |
| All | 10000 | 4.045 | 0.033 | 0.107 | 0.069 | 3.404 | 4.738 |

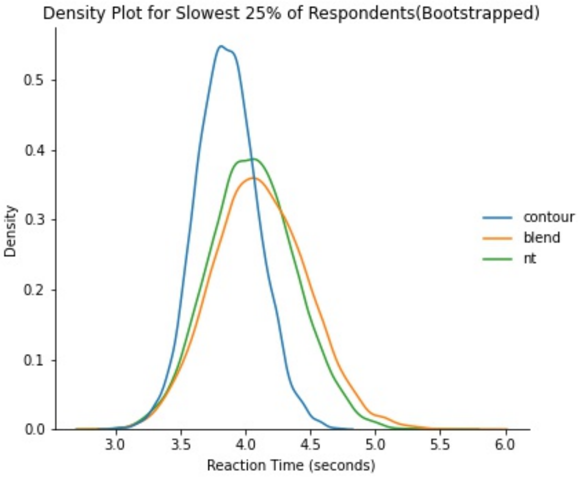

Figure 7: Density Plot of Results for slowest 25% of participants with a mean bootstrapping to 10,000 samples

## Participant Preference

The end of the study included an opinion question. The respondents were asked:

> During the course of the study, which style of image gave you the most confidence you were picking the correct area?

Figure 8 shows that 53% of participants felt that the *Contour* style attribution map gave them the most confidence in selecting the right area. The *Blended* option received 29.4% of the vote and *Raw SmoothGrad* received only 17.6%. This shows a strong association between the *Contour Attribution Map* and a level of confidence. The actual accuracy of responses was not measured and a future study to correlate the confidence with accuracy is recommended.

## Conclusion

The timing test results were inconclusive in determining a significant advantage of the proposed *Contour Attribution Map* method. It performed equally at worst and in the case of the slowest 25% of participants, it significantly reduced the variance in the responses and the overall maximum response time.

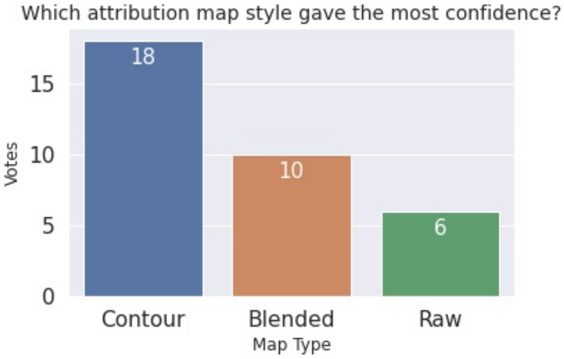

Figure 8: Confidence Survey Results

For the intended purposes in hard real-time systems, it could prove effective in minimizing the impact of a difficult attribution or a slower responding operator. The lack of clear-cut results warrants further investigation. From a systems engineering perspective, the favorable survey results and no decrease in performance would make *Contoured Attribution Maps* a favorable selection.

## Future Studies

After analyzing the results of this study, there are some key gaps in evaluating the attribution map algorithms. To better evaluate them another study in the future should implement the following:

- A hand-labeled data set that indicates the networks focus to allow for calculating actual accuracy.
- Larger participant pool with non-anonymous results to compare across professions, gender, age, etc.
- A larger image set shown to participants to allow for them to get the same picture in multiple formats and compare.
- Image sets which are carefully selected to try and introduce complexity and variance to the results.

Using some of the above would give a more complete understanding on the timing and accuracy of the human response to each of the attribution maps.

## References

Baehrens, D.; Schroeter, T.; Harmeling, S.; Kawanabe, M.; Hansen, K.; and Müller, K.-R. 2010. How to explain individual classification decisions. *The Journal of Machine Learning Research* 11:1803–1831.

Bien, N.; Rajpurkar, P.; Ball, R. L.; Irvin, J.; Park, A.; Jones, E.; Bereket, M.; Patel, B. N.; Yeom, K. W.; Shpanskaya, K.; Halabi, S.; Zucker, E.; Fanton, G.; Amanatullah, D. F.; Beaulieu, C. F.; Riley, G. M.; Stewart, R. J.; Blankenberg, F. G.; Larson, D. B.; Jones, R. H.; Langlotz, C. P.; Ng, A. Y.; and Lungren, M. P. 2018. Deep-learning-assisted diagnosis for knee magnetic resonance imaging: Development and retrospective validation of mrnet. *PLOS Medicine* 15(11):1–19.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Efron, B. 1992. Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics*. Springer. 569–593.

Erhan, D.; Bengio, Y.; Courville, A.; and Vincent, P. 2009. Visualizing higher-layer features of a deep network. *Technical Report, Univeristé de Montréal*.

Jain, A.; Bansal, R.; Kumar, A.; and Singh, K. 2015. A comparative study of visual and auditory reaction times on the basis of gender and physical activity levels of medical first year students. *International Journal of Applied and Basic Medical Research* 5(2):124.

Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps.

Smilkov, D.; Thorat, N.; Kim, B.; Viégas, F.; and Wattenberg, M. 2017. Smoothgrad: removing noise by adding noise.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. *arXiv preprint arXiv:1703.01365*.

# Appendices

The participant results:

Table 5: Full Timing Test Results

| | blended | nt | contour | folder | conf | average |
|---|---|---|---|---|---|---|
| 0 | [5.025, 3.161, 1.545, 2.246, 1.999, 1.866, 1.457] | [25.646, 1.649, 1.596, 2.363, 1.561, 0.912] | [5.847, 5.943, 1.294, 1.859, 1.134, 1.333, 0.968] | 0 | blended | 3.57263 |
| 1 | [20.877, 3.76, 3.63, 1.985, 3.924, 3.639] | [11.278, 2.985, 4.289, 2.788, 6.267, 3.824, 3.385] | [2.533, 3.32, 2.686, 2.045, 2.344, 4.066, 4.352] | 1 | contour | 4.77521 |
| 2 | [3.734, 2.843, 2.581, 1.498, 3.845, 5.96] | [3.117, 4.75, 11.91, 5.192, 2.102, 1.87, 3.785] | [23.145, 3.79, 2.388, 1.801, 2.437, 8.102, 3.416] | 1 | nt | 4.84172 |
| 3 | [0.456, 0.39, 0.387, 0.356, 0.362, 0.357] | [0.47, 0.545, 0.528, 0.396, 0.644, 0.418, 0.351] | [0.637, 0.363, 0.414, 1.161, 0.569, 0.376, 0.45] | 1 | blended | 0.476889 |
| 4 | [4.384, 6.955, 5.84, 2.208, 3.545, 2.916, 1.35] | [4.533, 1.61, 2.09, 6.295, 1.992, 4.036] | [2.479, 6.093, 3.359, 2.71, 1.756, 2.813, 1.437] | 0 | contour | 3.42033 |
| 5 | [6.636, 7.908, 10.682, 6.883, 8.343, 3.813] | [7.978, 6.737, 7.756, 10.59, 4.451, 6.549, 8.336] | [6.367, 6.232, 9.868, 4.363, 2.979, 4.829, 5.312] | 1 | blended | 6.85664 |
| 6 | [11.476, 12.241, 3.984, 1.792, 1.978, 1.825, 2.294] | [9.364, 1.947, 1.707, 1.329, 3.716, 4.884] | [8.948, 4.024, 4.827, 6.823, 3.642, 2.572, 1.485] | 2 | contour | 4.50869 |
| 7 | [2.603, 3.447, 2.042, 3.364, 4.292, 5.313] | [7.707, 2.95, 8.587, 4.139, 6.747, 2.895, 5.17] | [4.706, 3.744, 2.418, 3.677, 3.404, 4.903, 4.875] | 1 | contour | 4.3092 |
| 8 | [5.045, 9.192, 6.293, 1.889, 2.603, 8.93, 1.409] | [5.718, 4.898, 2.152, 1.426, 2.638, 14.746] | [6.457, 4.599, 2.198, 8.43, 4.673, 3.374, 8.523] | 2 | contour | 5.25981 |
| 9 | [4.81, 2.326, 2.144, 1.257, 1.809, 5.325, 1.326] | [3.376, 1.166, 1.419, 1.275, 1.674, 1.133] | [1.92, 1.877, 1.522, 1.364, 1.74, 2.564, 1.247] | 0 | contour | 2.04513 |
| 10 | [1.577, 1.629, 2.308, 1.69, 1.427, 1.406, 1.988] | [1.28, 1.326, 1.264, 0.96, 1.594, 2.111] | [1.939, 1.624, 1.494, 1.473, 1.16, 1.998, 1.285] | 2 | contour | 1.56931 |
| 11 | [3.751, 2.927, 2.307, 2.842, 2.286, 2.209] | [4.106, 2.24, 3.841, 3.416, 3.103, 1.785, 3.988] | [4.436, 2.351, 2.72, 1.956, 2.411, 4.328, 3.673] | 1 | contour | 3.01887 |
| 12 | [1.688, 1.194, 1.101, 1.091, 6.202, 2.381, 1.397] | [14.418, 3.03, 1.521, 2.146, 0.983, 3.484] | [3.083, 1.086, 1.051, 2.157, 2.53, 1.938, 1.468] | 0 | contour | 2.77203 |
| 13 | [2.062, 2.971, 2.861, 2.117, 2.964, 3.034, 2.449] | [2.958, 2.281, 2.711, 3.243, 3.157, 2.066] | [3.47, 2.583, 2.501, 4.111, 3.163, 1.773, 1.982] | 0 | contour | 2.72348 |
| 14 | [3.234, 1.384, 1.448, 1.552, 1.465, 1.379, 1.329] | [1.832, 1.317, 1.626, 1.578, 1.638, 2.506] | [1.885, 2.209, 1.439, 2.448, 1.524, 2.62, 1.986] | 2 | contour | 1.8166 |
| 15 | [3.42, 1.8, 2.437, 1.81, 1.091, 3.949, 1.468] | [1.816, 2.596, 2.898, 1.152, 1.109, 4.041] | [6.912, 4.852, 2.109, 1.718, 1.457, 1.709, 1.523] | 2 | contour | 2.48265 |
| 16 | [2.126, 2.87, 2.799, 2.232, 2.383, 3.111] | [1.624, 2.652, 2.177, 2.711, 2.06, 1.91, 2.05] | [2.406, 2.069, 1.417, 1.48, 1.737, 2.583, 2.668] | 1 | blended | 2.26913 |
| 17 | [2.775, 4.916, 2.409, 1.78, 5.247, 1.519, 1.905] | [2.228, 5.7, 4.498, 1.607, 3.33, 6.378] | [1.992, 5.036, 1.902, 6.198, 4.306, 4.367, 4.165] | 2 | blended | 3.62928 |
| 18 | [30.133, 1.24, 1.567, 1.608, 2.624, 1.899] | [4.479, 1.646, 1.896, 2.369, 1.739, 1.59, 1.587] | [36.729, 2.844, 2.27, 1.086, 1.98, 2.04, 2.902] | 1 | contour | 5.27333 |
| 19 | [1.284, 1.652, 1.951, 0.968, 1.136, 1.674, 0.919] | [2.635, 1.597, 1.203, 1.037, 1.94, 1.837] | [1.713, 0.994, 1.251, 1.343, 1.362, 1.35, 1.147] | 0 | contour | 1.46196 |
| 20 | [0.937, 1.07, 0.887, 0.858, 1.589, 2.916] | [1.715, 0.934, 1.377, 1.398, 0.914, 1.521, 1.756] | [0.95, 1.058, 1.052, 0.807, 0.962, 1.228, 1.252] | 1 | contour | 1.26463 |
| 21 | [0.995, 1.816, 1.52, 1.673, 1.712, 1.918, 1.172] | [1.719, 1.217, 1.805, 2.306, 2.488, 1.571] | [1.436, 1.265, 1.394, 1.072, 1.966, 2.774, 1.485] | 0 | blended | 1.67405 |
| 22 | [3.619, 3.595, 6.305, 2.243, 2.811, 3.572, 1.403] | [10.165, 1.629, 3.615, 1.682, 4.46, 1.925] | [1.592, 1.73, 1.644, 1.532, 1.809, 2.738, 3.717] | 0 | blended | 3.12851 |
| 23 | [7.83, 3.222, 7.916, 1.394, 1.933, 3.785, 2.336] | [1.992, 2.682, 3.688, 1.321, 2.644, 3.181] | [11.971, 5.86, 2.013, 2.614, 3.936, 4.632, 3.634] | 2 | blended | 3.86517 |
| 24 | [2.172, 2.642, 1.335, 2.043, 3.193, 4.094] | [2.158, 2.49, 2.005, 1.803, 1.726, 1.728, 2.224] | [2.764, 2.115, 1.525, 1.633, 1.294, 1.743, 2.22] | 1 | nt | 2.16604 |
| 25 | [2.099, 1.983, 1.963, 1.563, 1.726, 1.41] | [1.873, 2.258, 2.126, 2.181, 1.523, 1.497, 2.045] | [1.916, 1.97, 1.896, 1.765, 1.59, 1.437, 2.761] | 1 | nt | 1.87489 |
| 26 | [3.401, 1.671, 1.638, 2.086, 1.635, 1.672, 1.708] | [2.196, 1.699, 1.621, 1.493, 1.616, 2.956] | [5.385, 1.538, 1.611, 2.256, 2.47, 3.03, 2.229] | 2 | nt | 2.18291 |
| 27 | [2.37, 1.755, 2.037, 1.705, 2.05, 1.957, 2.01] | [2.465, 2.017, 2.161, 3.479, 2.102, 1.805] | [1.588, 1.639, 1.524, 2.885, 1.463, 1.904, 2.872] | 0 | blended | 2.10125 |
| 28 | [2.695, 2.592, 0.973, 1.091, 1.203, 1.318] | [1.714, 4.455, 3.152, 2.17, 1.494, 1.55, 1.169] | [9.106, 2.826, 1.701, 1.36, 1.196, 1.132, 1.511] | 1 | nt | 2.19302 |
| 29 | [1.95, 2.339, 1.521, 1.628, 2.463, 3.722, 2.925] | [12.205, 3.791, 2.604, 2.307, 2.91, 2.218] | [2.654, 2.287, 5.55, 2.365, 11.622, 5.128, 1.586] | 0 | blended | 3.71972 |
| 30 | [1.042, 1.164, 1.51, 1.58, 2.36, 2.055] | [1.766, 2.585, 4.354, 1.766, 1.381, 1.454, 1.694] | [1.337, 1.505, 1.681, 1.594, 1.272, 3.441, 1.81] | 1 | contour | 1.85569 |
| 31 | [1.344, 1.398, 1.755, 1.1, 1.696, 1.564] | [6.652, 1.33, 4.488, 1.388, 1.415, 1.372, 2.696] | [2.079, 1.429, 1.341, 1.533, 1.335, 1.809, 2.298] | nan | contour | 1.9761 |
| 32 | [5.032, 11.051, 3.164, 2.224, 2.518, 2.427, 1.704] | [20.267, 2.561, 2.31, 1.881, 1.59, 1.887] | [5.268, 2.259, 2.128, 2.831, 2.67, 3.034, 1.975] | nan | contour | 3.99351 |
| 33 | [1.958, 1.988, 1.092, 0.986, 1.247, 0.913] | [2.029, 1.735, 2.859, 3.003, 0.967, 1.17, 1.89] | [2.358, 1.577, 1.249, 1.295, 1.024, 1.313, 1.437] | nan | nt | 1.59305 |