# Doctor, Is This Normal?

ULAB, Division of Totally Serious Science

July 21, 2021

## 1 Introduction

Around May 2021, a humorous image of a sidewalk appeared on various social-medial platforms. The image, Figure 1 below, claims that a sidewalk crack resembles the form of a normal distribution. In this report, we will test this claim by fitting a standard normal curve.


Figure 1: Internet image.

## 2 Methods

### 2.1 Preprocessing

First, we use the Perspective Warp tool in Photoshop to correct the titled perspective of the image. Next, our analysis shifts to Python. Using PIL and NumPy[1], the image is converted into grayscale.

A threshold mask is then applied to the image so that only dark pixels would be retained. Note that some minimal manual processing was required to remove the vertical sidewalk crack as well as outlier pixels.

Finally, the location of the dark pixels are recorded as coordinates in a Cartesian plane.

### 2.2 Curve Fitting

The SciPy package[2] was used to fit a normal distribution,

$$f(x) = a \cdot e^{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma}} \tag{1}$$

where $a$ is a scaling constant, $\mu$ is mean, and $\sigma$ is the standard deviation.

## 3 Results

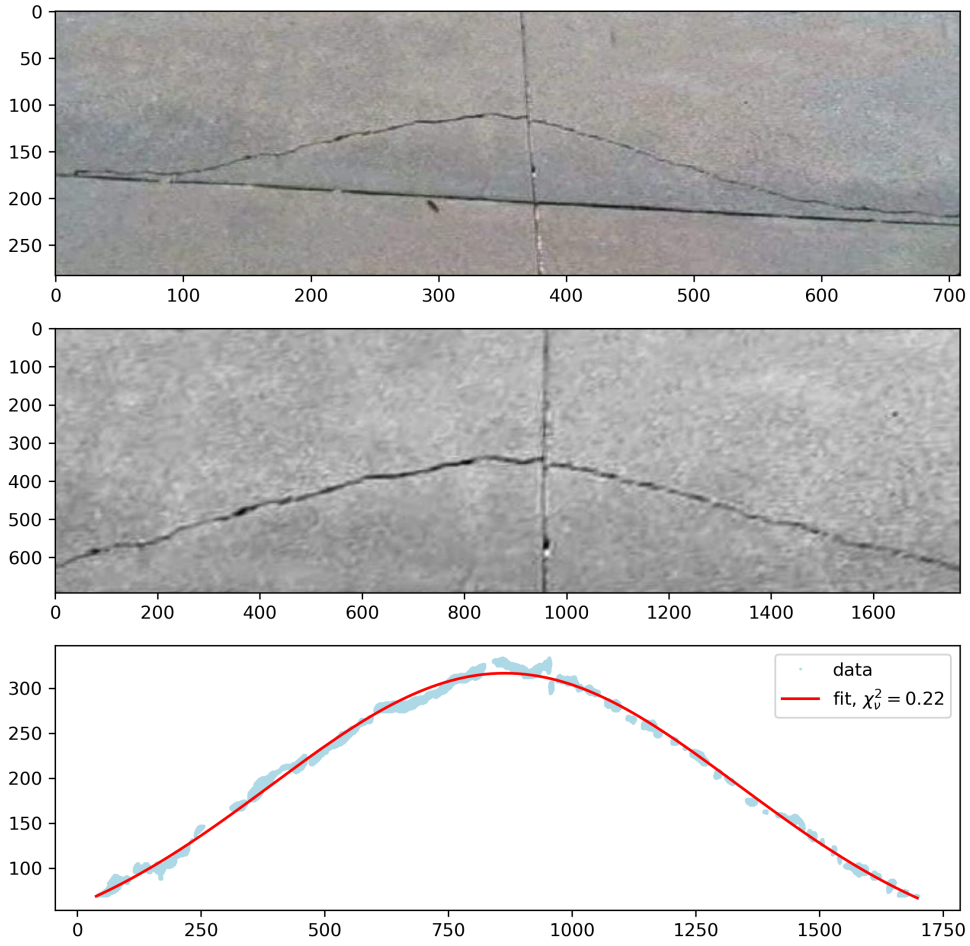


Figure 2: (a) original image, (b) processed image with perspective correction and grayscale transformation, (c) curve fit of the sidewalk crack points.

Visually (Figure 2), the cracks fits well with a normal distribution. Furthermore, we determined the fit to have a reduced chi-squared value,

$$\chi^2_\nu = \frac{1}{\nu}\sum\frac{(O-E)^2}{E} = 0.22 \tag{2}$$

where $\nu$ is the number of degrees of freedom. (Note: we used the Pearson chi-squared definition because it is difficult to estimate the uncertainty of our points.)

The reduced chi-squares is less than 1 and small. This means that our model is likely correct but that the data is slightly over fitted. Regardless of the latter point, we have shown that the crack is well-described by a normal curve!

# References


[1] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. Fernández del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, p. 357–362, 2020.

[2] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.