Consider the steady heat conduction problem that was introduced in class:

$$\begin{aligned} \nabla \cdot (k(x, \boldsymbol{\xi}) \nabla u(\boldsymbol{x})) &= 0\,, & x &\in D\,, \\ u(\boldsymbol{x}) &= 0\,, & \boldsymbol{x} &\in \Gamma_{\mathrm{top}}\,, \\ \nabla u(\boldsymbol{x}) \cdot n &= 0\,, & \boldsymbol{x} &\in \Gamma_{\mathrm{side}}\,, \\ \nabla u(\boldsymbol{x}) \cdot n &= 1\,, & \boldsymbol{x} &\in \Gamma_{\mathrm{base}}\,, \end{aligned}$$

where $D = (0,1)^2 \subset \mathbb{R}^2$. The quantity of interest is

$$Q(u) = \int_{\Gamma_{\mathrm{base}}} u(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}\,,$$

and the conductivity coefficient is

$$k(\boldsymbol{x}, \boldsymbol{\xi}) = \begin{cases} \xi_1\,, & 0 \le x_1 < \frac{1}{4} \text{ and } 0 \le x_2 < \frac{1}{4} \\ \vdots & \\ \xi_{16}\,, & \frac{3}{4} \le x_1 \le 1 \text{ and } \frac{3}{4} \le x_2 \le 1\,, \end{cases}$$

Let the vector \xi = [\xi_1, ..., \xi_{16}] be in the domain [0.1, 10]^16. Learn a map with a deep network from \xi to Q(u) where u() is the solution corresponding to the vector \xi.

Use the following matlab code to generate training data

[redacted]

Generate a plot that tests your learned deep network on uniformly sampled \xi vectors in [0.1, 10]^16 and compare (relative error) against the “truth” computed with the Matlab code.