GPs and differentiation

Three things are attempted below, each presented in their own section:

Section 1: How to fit a GP to an unknown function.

Section 2: How the GP representation of a derivative of an unknown function represented by a GP is related to the GP representation of the function.

Section 3: How to compare the alignment of two surfaces utilising dot products of their partial derivative vectors.

1 Fitting a GP in `R`

The following is taken from Keirstead (2012), a demo of Gaussian process regression with R by James Keirstead (5 April 2012). He writes:

> Chapter 2 of Rasmussen and Williams (2006) provides a detailed explanation of the math for Gaussian process regression. It doesn't provide much in the way of code though. This Gist is a brief demo of the basic elements of Gaussian process regression, as described on pages 13 to 16.

1.1 The GP prior

First, for a (dense) set of 50 regularly-spaced input values (denoted x^*) we populate the covariance matrix using the chosen covariance function, or *kernel*, $k(\cdot,\cdot)$, in this case the squared exponential with length parameter $\Psi = 1$. One such covariance matrix $k(x^*, x^*)$ is printed below, but for only 10 regularly-spaced points, rather than the 50 used throughout the rest of the work:

$$\begin{bmatrix}
1 & 0.784 & 0.377 & 0.112 & 0.02 & 0.002 & 0 & 0 & 0 & 0 \\
0.784 & 1 & 0.784 & 0.377 & 0.112 & 0.02 & 0.002 & 0 & 0 & 0 \\
0.377 & 0.784 & 1 & 0.784 & 0.377 & 0.112 & 0.02 & 0.002 & 0 & 0 \\
0.112 & 0.377 & 0.784 & 1 & 0.784 & 0.377 & 0.112 & 0.02 & 0.002 & 0 \\
0.02 & 0.112 & 0.377 & 0.784 & 1 & 0.784 & 0.377 & 0.112 & 0.02 & 0.002 \\
0.002 & 0.02 & 0.112 & 0.377 & 0.784 & 1 & 0.784 & 0.377 & 0.112 & 0.02 \\
0 & 0.002 & 0.02 & 0.112 & 0.377 & 0.784 & 1 & 0.784 & 0.377 & 0.112 \\
0 & 0 & 0.002 & 0.02 & 0.112 & 0.377 & 0.784 & 1 & 0.784 & 0.377 \\
0 & 0 & 0 & 0.002 & 0.02 & 0.112 & 0.377 & 0.784 & 1 & 0.784 \\
0 & 0 & 0 & 0 & 0.002 & 0.02 & 0.112 & 0.377 & 0.784 & 1
\end{bmatrix}$$

Figure 1 plots some sample functions drawn from the the zero-mean Gaussian process prior with the aforementioned covariance matrix in order to give an idea of the type of functions it specifies.

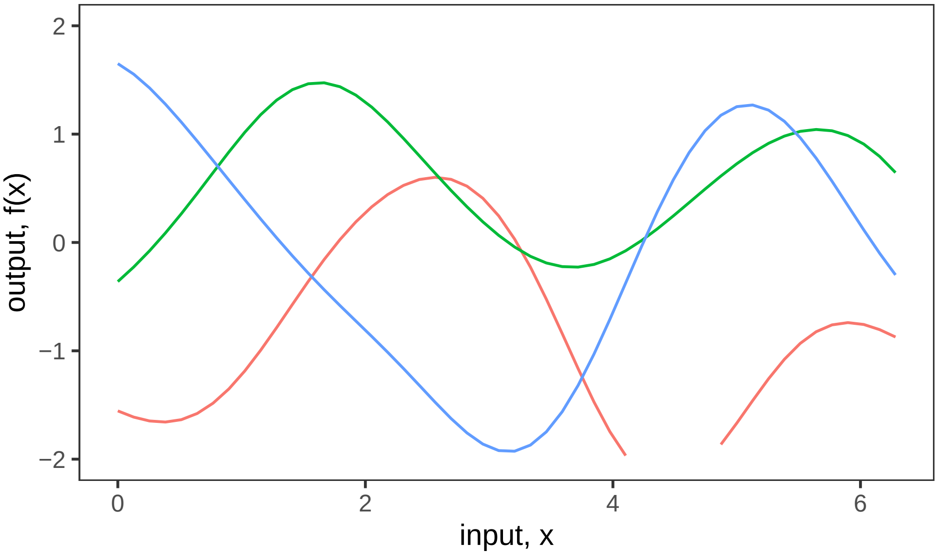

Figure 1: Three functions sampled from the GP prior distribution.

1.2 GP posteriors

Next, we'll generate 5 data points using the $\sin(x)$ function in the range $x \in [0.2, 2\pi]$, imagining that that is the underlying real-like function we're interested in learning about. An observation error, randomly sampled from $\mathrm{N}(0, 0.1^2)$, is added to each term. The 'observed' values are denoted z and the associated input values x. The observations produced are shown in Figure 2, and the observation error added on visible by the fact that they tend not to sit exactly on the function.

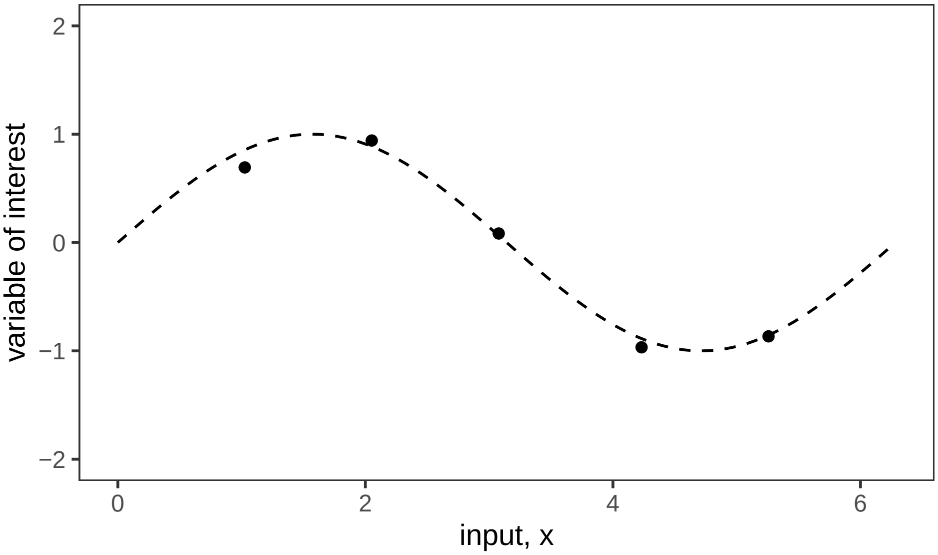

Figure 2: The true process, $\zeta(x)$, is indicated with the black dashed line, and five observations, z, by the points.

1.2.1 Noisy-observations but a noise-free GP posterior

To specify the posterior GP distribution, we need three further covariance matrices: $k(x, x)$, $k(x, x^*)$ and $k(x^*, x)$ (recall that $k(\cdot, \cdot)$ was described in Section 1.1). Using these four covariance matrices, the posterior distribution is derived using Equation (2.19) in Rasmussen and Williams (2006):

$$\begin{aligned}\mathbf{f}^*|x^*, x, \mathbf{f} \sim N(&k(x^*, x)k(x, x)^{-1}\mathbf{y}, \\ &k(x^*, x^*) - k(x^*, x)k(x, x)^{-1}k(x, x^*))\end{aligned}$$

Next, we'll generate and plot 20 functions from the posterior distribution, along with the mean function and 95% confidence interval (Figure 3). Outside of the interval on which we have data, the mean function returns towards zero.

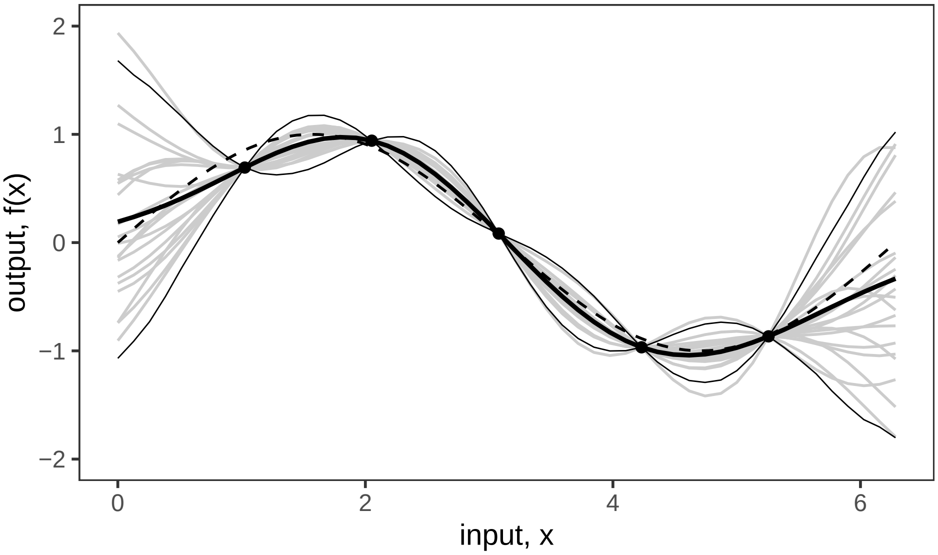

Figure 3: The posterior mean function (thick black solid line) along with 20 functions sampled from the posterior (grey) and 95% confidence intervals (thin black solid lines). The underlying real-life function is again a dashed black line.

1.2.2 Noisy-observations and a GP posterior with noise

Since there is observation error, it might make more sense for the posterior draws *not* to pass through the observed points. This can be achieved by adding a constant (observation noise) term onto the diagonal of the covariance matrix. Recalling that normally distributed measurement errors sampled from N(0, 0.01) were added to the simulated values, we'll add 0.1 to the diagonal of the covariance matrix as observation noise.

Figure 4 replicates Figure 3 to incorporate this added noise, and now the mean function (thick red line) doesn't pass through the data points, and clearly the posterior uncertainty (thin solid red lines) has increased:

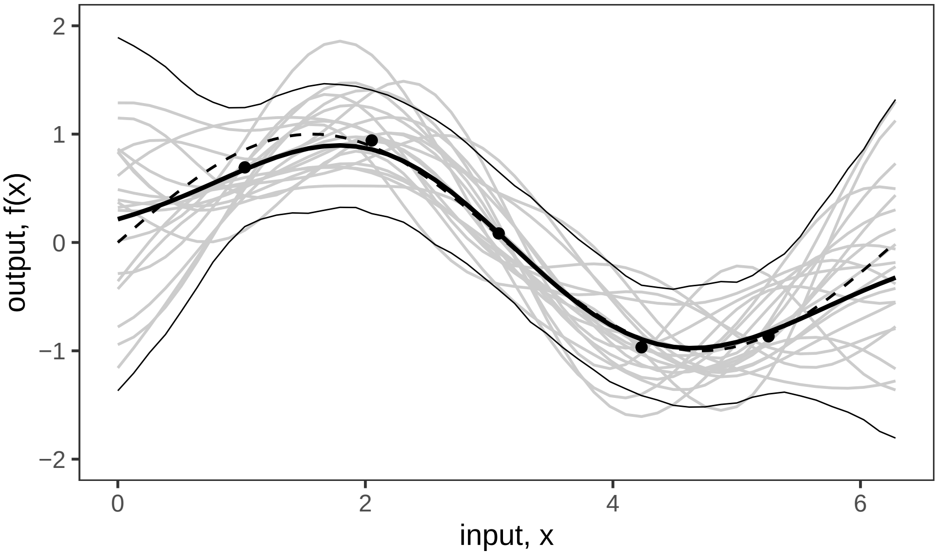

Figure 4: The posterior mean function (thick black solid line) along with 20 functions sampled from the posterior (grey) and 95% confidence intervals (thin black solid lines). The underlying real-life function is again a dashed black line.

The 95% confidence intervals have clearly changed between Figures 3 and 4. The mean function has slightly changed too – see Figure 5 – since Equation 1, which gives a closed-form expression for the mean function, includes (via $\boldsymbol{\alpha}$) involves $k(x, x)$, which is changed when including a nugget versus not including one.

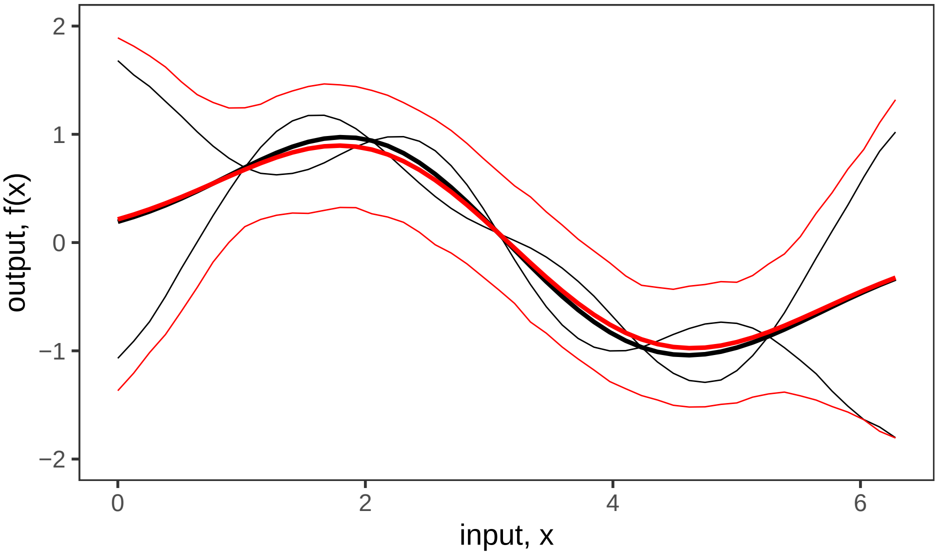

Figure 5: Using noisy observations for both, we see the posterior mean function from the GP posterior without and with noise (thick black and red lines respectively), along with corresponding 95% confidence intervals (thin black and red lines respectively). These are derived in Sections 1.2.1 and 1.2.2 respectively.

2 The GP representation of a derivative of an unknown function that is represented by a GP

Having fitted a GP to the data, and obtained a posterior mean function (or rather two) that lies close to the real-life underlying function, we turn our attention to the GP representation of a derivative of an unknown function that is represented by a GP. The GP posterior with noise derived in Section 1.2.2 will be used.

To obtain the derivative of the posterior mean function, we first need an expression for the posterior mean function. Viewing the posterior mean function, $\bar{f}(x^*)$, as a linear combination of 5 kernel functions, each centered at one of the 5 training points, Equation (2.27) in Rasmussen and Williams (2006), states that, for any particular input value x^*,

$$\bar{f}(x^*) = \sum_{i=1}^{5} a_i k(x_i, x^*) \tag{1}$$

where $\boldsymbol{\alpha} = (k(x,x) + 0.1I_5)^{-1}\mathbf{y}$. As such, the posterior mean function here will be

$$\begin{aligned} \bar{f}(x^*) = \alpha_1 \exp\left(-\frac{1}{2}\left(\frac{x_1 - x^*}{l}\right)^2\right) + \alpha_2 \exp\left(-\frac{1}{2}\left(\frac{x_2 - x^*}{l}\right)^2\right) \\ + \alpha_3 \exp\left(-\frac{1}{2}\left(\frac{x_3 - x^*}{l}\right)^2\right) + \alpha_4 \exp\left(-\frac{1}{2}\left(\frac{x_4 - x^*}{l}\right)^2\right) + \alpha_5 \exp\left(-\frac{1}{2}\left(\frac{x_5 - x^*}{l}\right)^2\right), \end{aligned} \tag{2}$$

which is plotted as the solid black line in Figure 6.

From O'Hagan (1992), the derivatives of functions modelled by a Gaussian process with

$$\begin{aligned} E\left\{\eta(x)\right\} &= \mathbf{h}^T(x)\boldsymbol{\beta} \\ Cov\left\{\eta(x), \eta(\mathbf{x}')\right\} &= k(x,x) \end{aligned}$$

can also be modelled by a Gaussian process, with

$$\begin{aligned} E\left\{\frac{\partial}{\partial x}\eta(x)\right\} &= \frac{\partial}{\partial x}\mathbf{h}^T(x)\boldsymbol{\beta} \\ Cov\left\{\frac{\partial}{\partial x}\eta(x), \frac{\partial}{\partial x}\eta(x')\right\} &= \frac{\partial^2}{\partial x \partial x'}k(x,x'). \end{aligned} \tag{3}$$

Led by Equation (2), differentiating the posterior mean function in Equation (1) gives

$$\frac{\partial}{\partial x^*}\bar{f}(x^*) = \frac{1}{l^2}\left[\alpha_1(x_1 - x^*)\exp\left(-\frac{1}{2}\left(\frac{x_1 - x^*}{l}\right)^2\right) + \cdots + \alpha_5(x_5 - x^*)\exp\left(-\frac{1}{2}\left(\frac{x_5 - x^*}{l}\right)^2\right)\right],$$

which is also plotted in Figure 6 (solid red line). The real, underlying function $\sin(x)$, and its derivative $\cos(x)$ are included in Figure 6 too (black and red dashed lines respectively) for comparison.

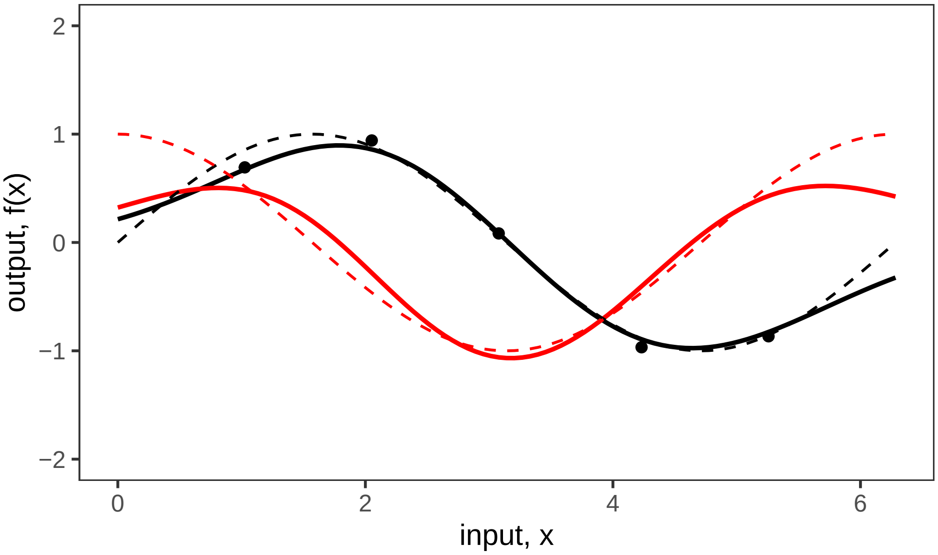

Figure 6: The posterior mean function (black solid line) and the true real-life process function, $\sin(x)$ (black dashed line). Also shown is the derivative of the posterior mean function (red solid line) and the derivative of the true real-life process function, $\cos(x)$ (red dashed line). Observations are shown as black dots.

With only five observations, both the posterior mean function of the unknown function and that of the derivative of the unknown function lie close to their true counterparts in the region in which the data lie (and less so outside of this range).

3 How to compare the alignment of two gradient fields

To define the comparison measure, we'll work through the steps ① - ⑦ outlined in the correspondence between Jonathan and Manolis.

① Computation of partial derivatives w.r.t. the parameters used the test the case.

The function against which the others will be compared is:

$$f(x) = \sqrt{100 - x_1^2 - x_2^2}$$

which has the following partial derivatives, with slope field and contour plot shown in Figure 7:

$$\frac{\partial f(x)}{\partial x_1} = -\frac{x_1}{\sqrt{100 - x_1^2 - x_2^2}}$$
$$\frac{\partial (x)}{\partial x_2} = -\frac{x_2}{\sqrt{100 - x_1^2 - x_2^2}}$$

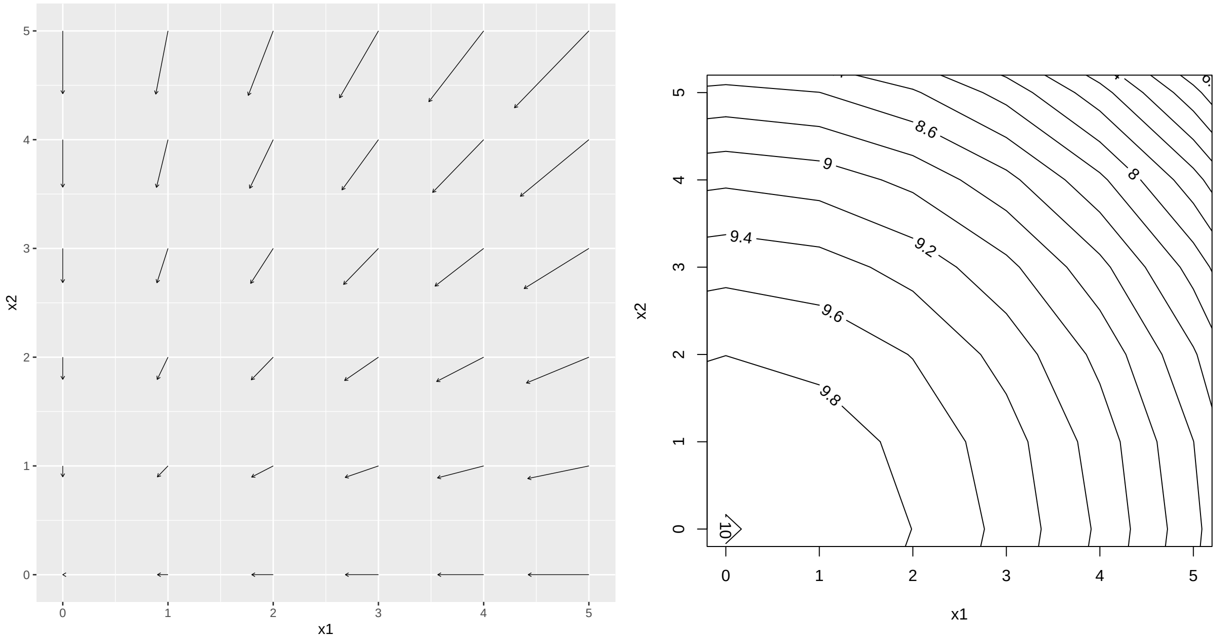

Figure 7: Slope field and contour plot for the function $f(x) = \sqrt{100 - x_1^2 - x_2^2}$

The relative values of the partial derivatives (w.r.t. x_1 and x_2) at 36 points can be inferred from the slope field in Figure 7.

The remaining six steps require the selection of another function (which we'll call $g(x)$) to compare to $f(x)$ from Step 1. First we'll look at choices for this second function which offer an acceptable match; initially one which is a perfect match.

3.1 Good agreement

3.1.1 Exact match

② Repeat Step 1 for the second response surface.

Since we're matching $g(x)$ with $f(x)$, we have

$$g(x) = \sqrt{100 - x_1^2 - x_2^2}$$

which has the following partial derivatives, with slope field and contour plot shown in Figure 8:

$$\frac{\partial g(x)}{\partial x_1} = -\frac{x_1}{\sqrt{100 - x_1^2 - x_2^2}}$$
$$\frac{\partial g(x)}{\partial x_2} = -\frac{x_2}{\sqrt{100 - x_1^2 - x_2^2}}$$

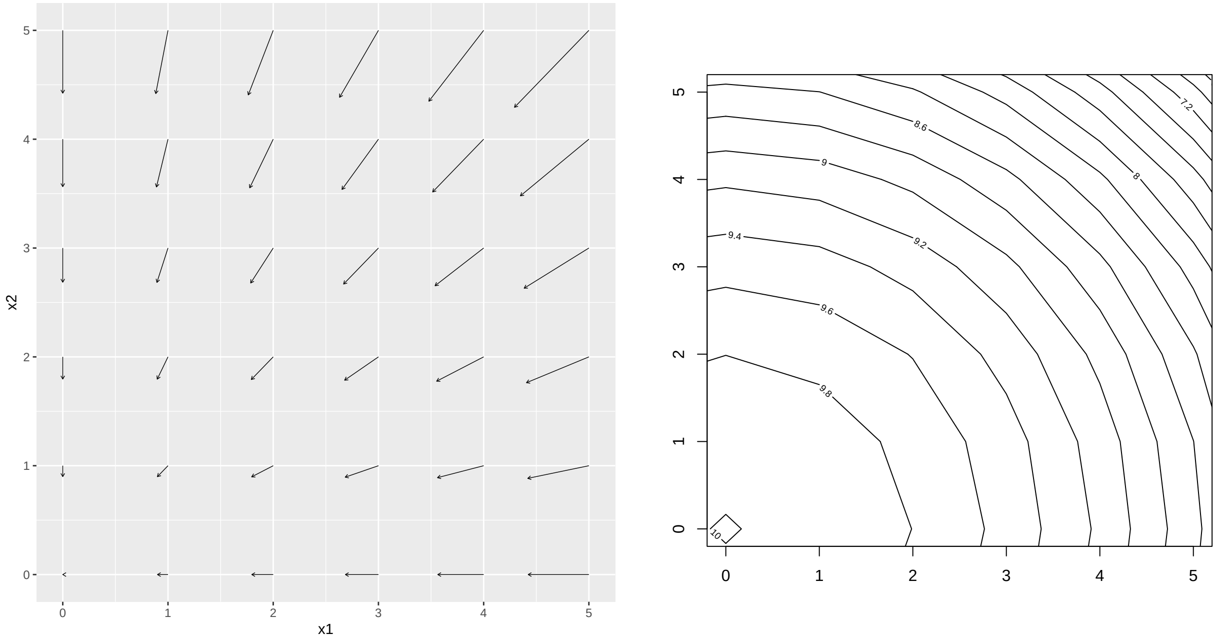

Figure 8: Slope field and contour plot for the function $g(x) = \sqrt{100 - x_1^2 - x_2^2}$

③ Normalise vectors of partial derivatives.

This is done, and a slope field showing these normalised vectors is shown in Figure 9.

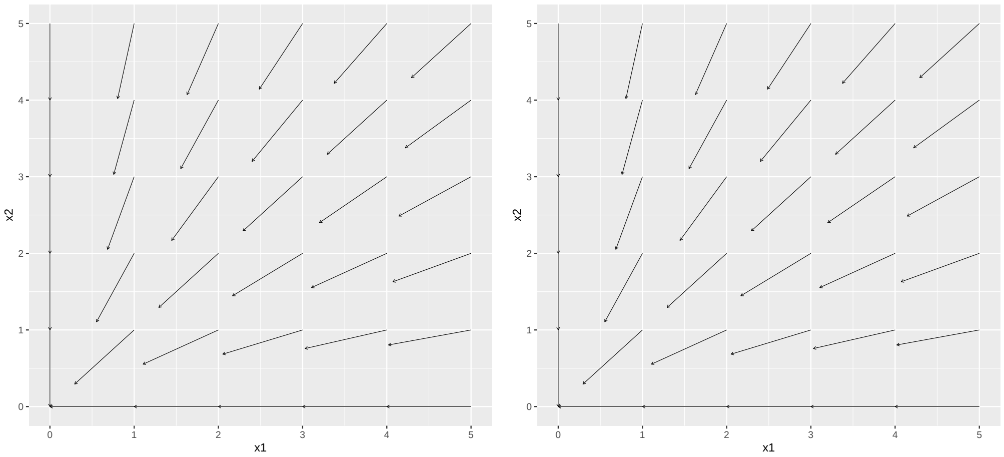

Figure 9: Normalised slope fields for the functions $f(x) = \sqrt{100 - x_1^2 - x_2^2}$ and $g(x) = \sqrt{100 - x_1^2 - x_2^2}$.

Steps ④-⑥ are performed together, and the result of Step ⑥ is discussed in Step ⑦.

④ Compute the dot product and take the absolute values.

⑤ Repeat the process over the entire parameter space.

⑥ Sum all together and divide by the number of discrete points.

⑦ Check how close this number is to 1.

For functions $f(x)$ and $g(x)$, whose contours plots are reprinted in Figure 10 below, we get a number of **0.97**. This is extremely close to 1, which is want we would expect, since the contours match exactly. The only reason this is not exactly 1 is that at $(0, 0)$, the dot product involves a (in fact two) zero vectors.

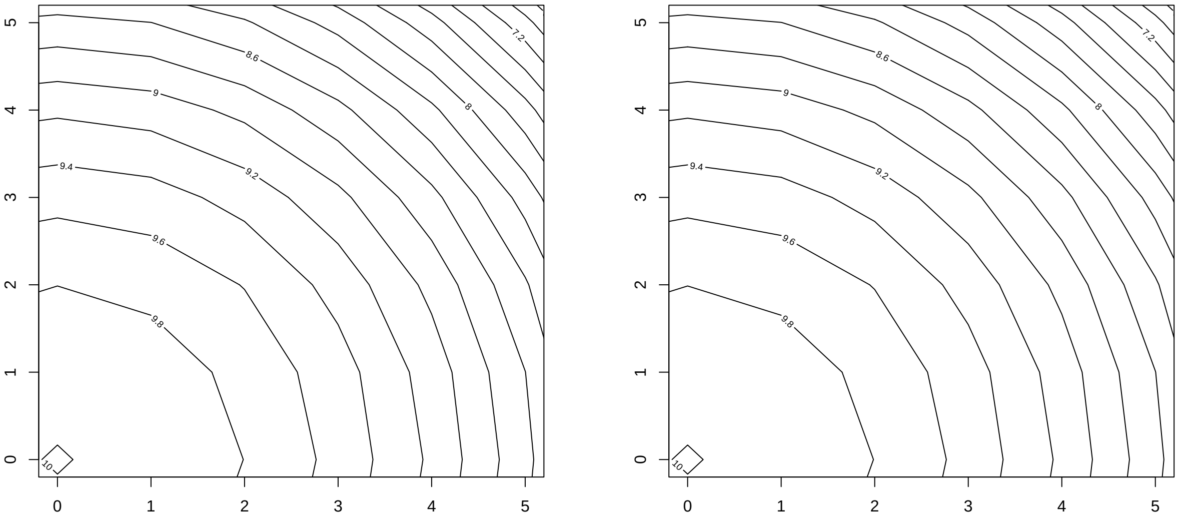

Figure 10: Contour plots for the functions $f(x) = \sqrt{100 - x_1^2 - x_2^2}$ and $g(x) = \sqrt{100 - x_1^2 - x_2^2}$.

3.2 Bad agreement

3.2.1 First try

② Repeat Step 1 for the second response surface.

$g(x)$ now takes the form:

$$g(x) = \sqrt{100 - (x_1 - 5)^2 - x_2^2}$$

which has the following partial derivatives, with slope field and contour plot shown in Figure 11:

$$\begin{aligned}\frac{\partial g(x)}{\partial x_1} &= -\frac{x_1 - 5}{\sqrt{100 - (x_1 - 5)^2 - x_2^2}} \\ \frac{\partial g(x)}{\partial x_2} &= -\frac{x_2}{\sqrt{100 - (x_1 - 5)^2 - x_2^2}}\end{aligned}$$

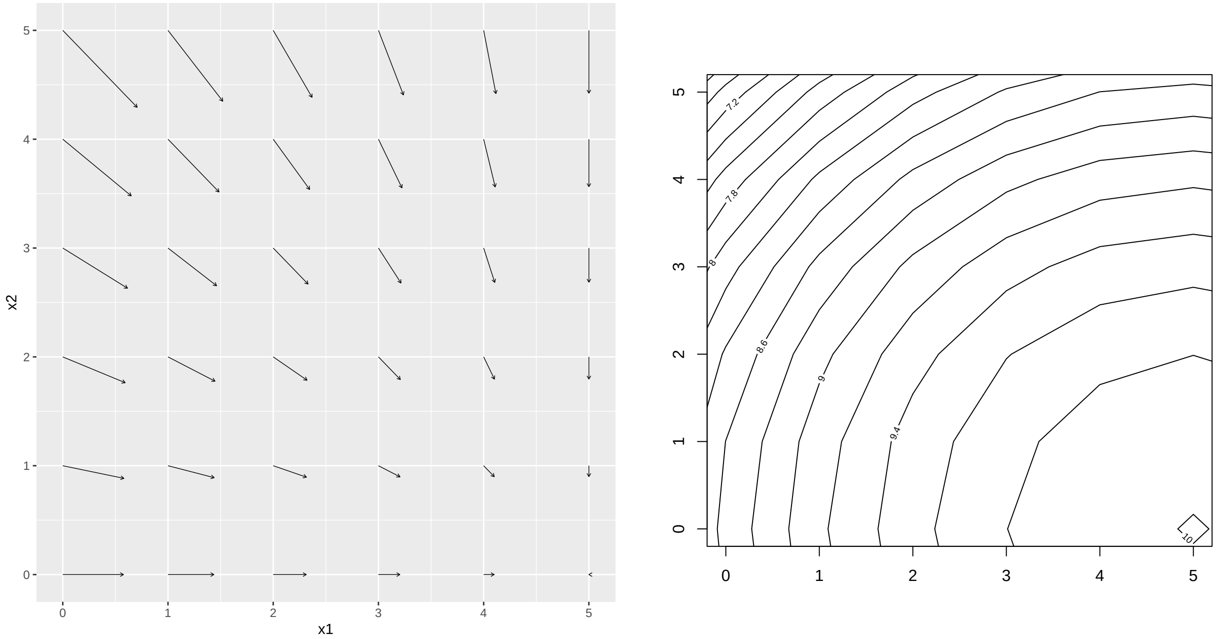

Figure 11: Slope field and contour plot for the function $g(x) = \sqrt{100 - (x_1 - 5)^2 - x_2^2}$

③ Normalise vectors of partial derivatives.

This is done, and a slope field showing these normalised vectors is shown in Figure 12.

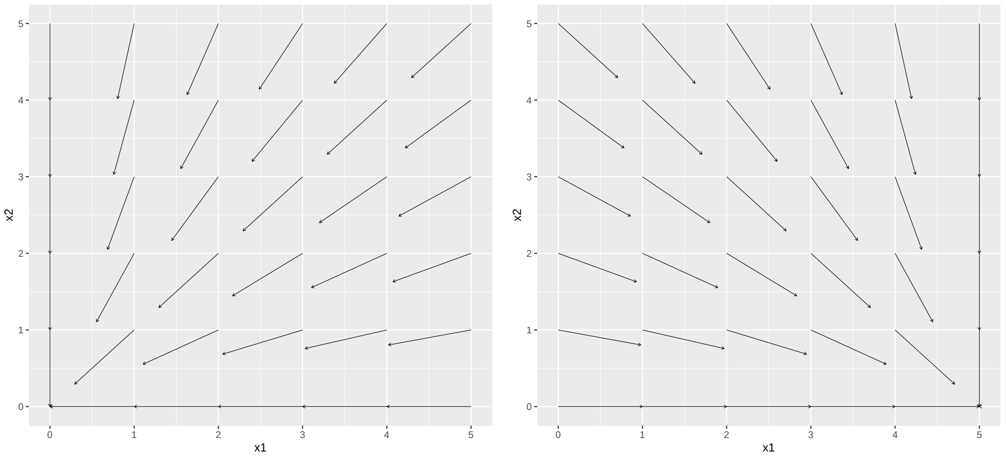

Figure 12: Normalised slope fields for the functions $f(x) = \sqrt{100 - x_1^2 - x_2^2}$ and $g(x) = \sqrt{100 - (x_1 - 5)^2 - x_2^2}$.

Steps ④-⑥ are performed together, and the result of Step ⑥ is discussed in Step ⑦.

④ Compute the dot product and take the absolute values.

⑤ Repeat the process over the entire parameter space.

⑥ Sum all together and divide by the number of discrete points.

⑦ Check how close this number is to 1.

For functions $f(x)$ and $g(x)$, whose contours plots are reprinted in Figure 13 below, we get a number of **0.48**.

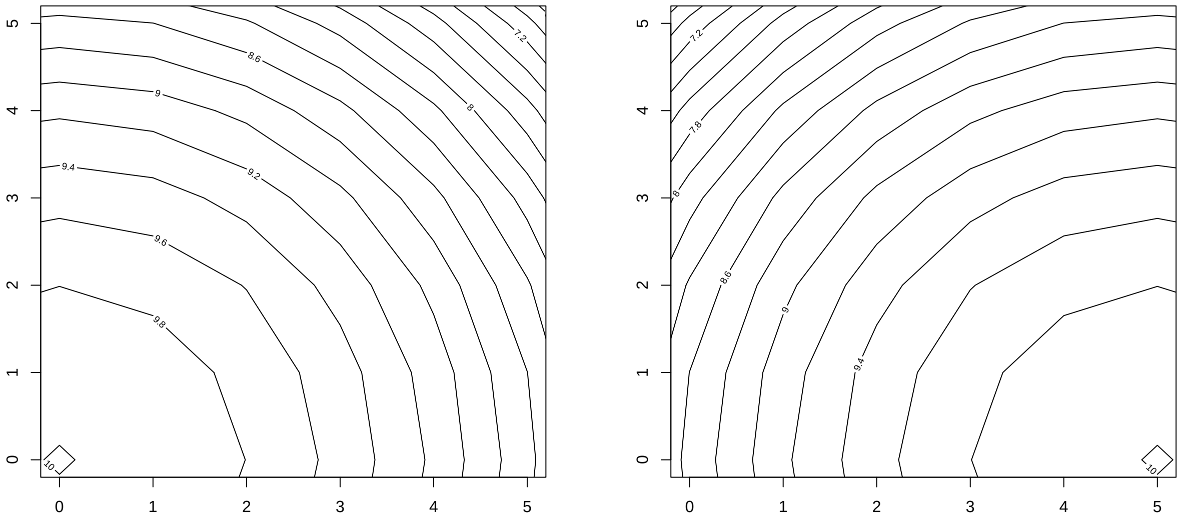

Figure 13: Contour plots for the functions $f(x) = \sqrt{100 - x_1^2 - x_2^2}$ and $g(x) = \sqrt{100 - (x_1 - 5)^2 - x_2^2}$.

3.2.2 Second try

② Repeat Step 1 for the second response surface.

$g(x)$ now takes the form:

$$g(x) = \sqrt{100 - x_1^2 - (x_2 - 5)^2}$$

which has the following partial derivatives, with slope field and contour plot shown in Figure 14:

$$\frac{\partial g(x)}{\partial x_1} = -\frac{x_1}{\sqrt{100 - x_1^2 - (x_2 - 5)^2}}$$
$$\frac{\partial g(x)}{\partial x_2} = -\frac{x_2 - 5}{\sqrt{100 - x_1^2 - (x_2 - 5)^2}}$$

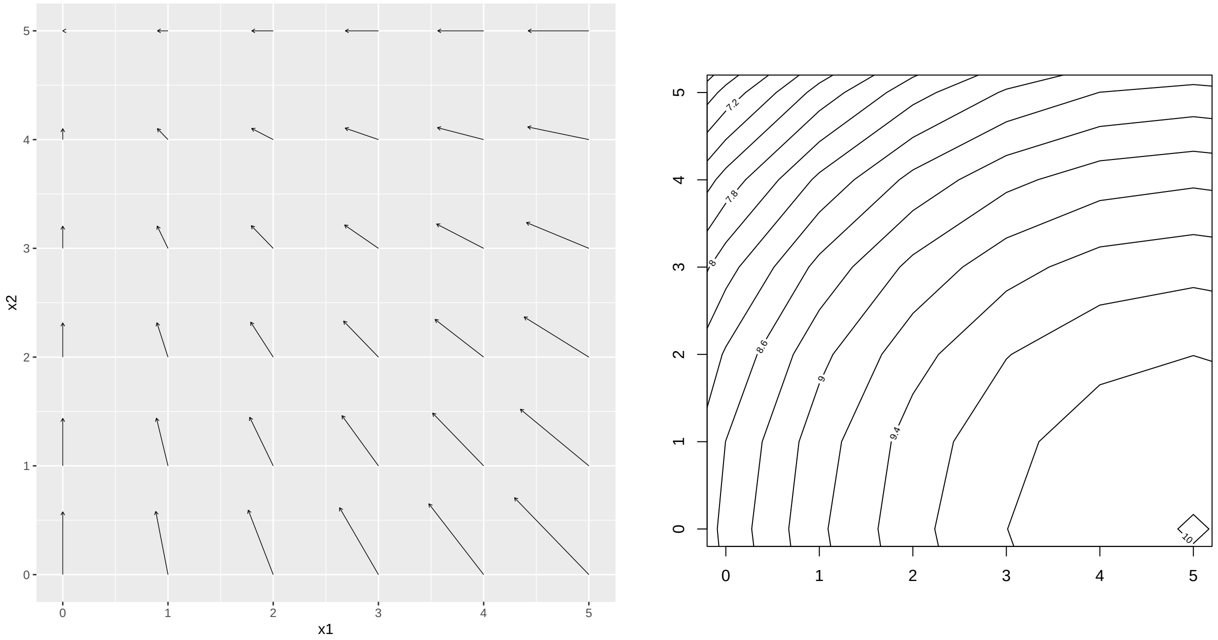

Figure 14: Slope field and contour plot for the function $g(x) = \sqrt{100 - x_1^2 - (x_2 - 5)^2}$

③ Normalise vectors of partial derivatives.

This is done, and a slope field showing these normalised vectors is shown in Figure 15.

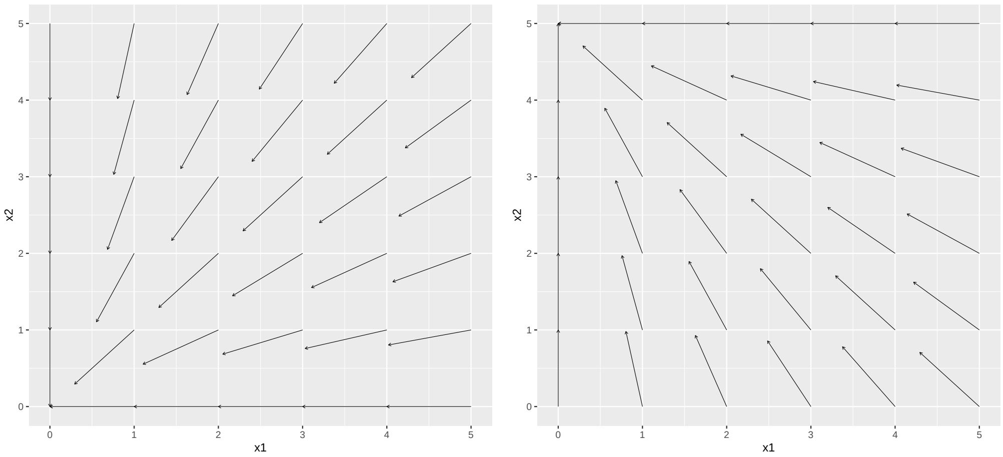

Figure 15: Normalised slope fields for the functions $f(x) = \sqrt{100 - x_1^2 - x_2^2}$ and $g(x) = \sqrt{100 - x_1^2 - (x_2 - 5)^2}$.

Steps ④-⑥ are performed together, and the result of Step ⑥ is discussed in Step ⑦.

④ Compute the dot product and take the absolute values.

⑤ Repeat the process over the entire parameter space.

⑥ Sum all together and divide by the number of discrete points.

⑦ Check how close this number is to 1.

For functions $f(x)$ and $g(x)$, whose contours plots are reprinted in Figure 16 below, we get a number of **0.48**.

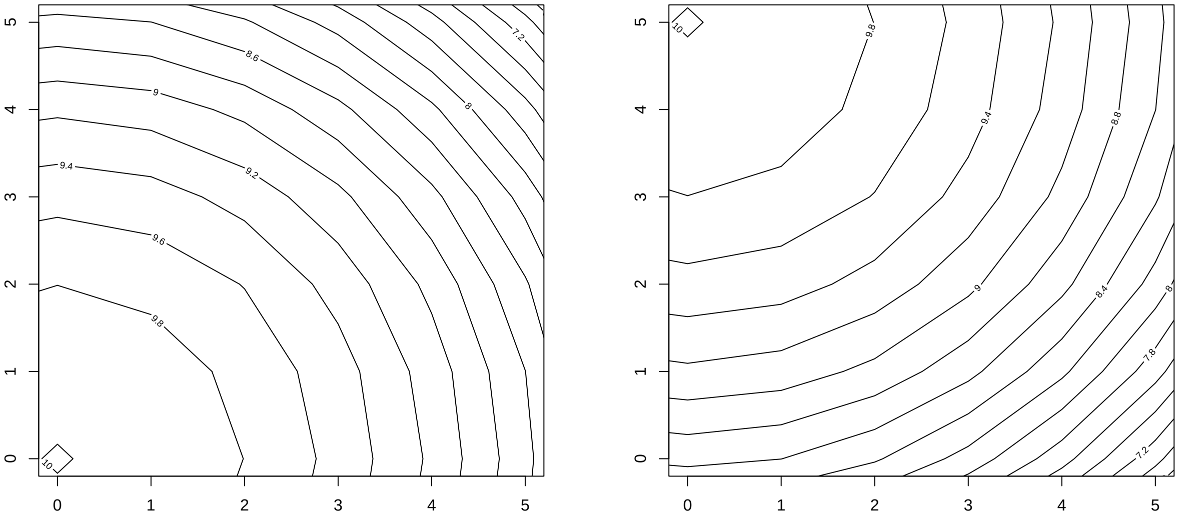

Figure 16: Contour plots for the functions $f(x) = \sqrt{100 - x_1^2 - x_2^2}$ and $g(x) = \sqrt{100 - x_1^2 - (x_2 - 5)^2}$.

3.2.3 Third try

② Repeat Step 1 for the second response surface.

$g(x)$ now takes the form:

$$g(x) = \sqrt{100 - (x_1 - 5)^2 - (x_2 - 5)^2}$$

which has the following partial derivatives, with slope field and contour plot shown in Figure 17:

$$\frac{\partial g(x)}{\partial x_1} = -\frac{x_1}{\sqrt{100 - (x_1 - 5)^2 - (x_2 - 5)^2}}$$
$$\frac{\partial g(x)}{\partial x_2} = -\frac{x_2 - 5}{\sqrt{100 - (x_1 - 5)^2 - (x_2 - 5)^2}}$$

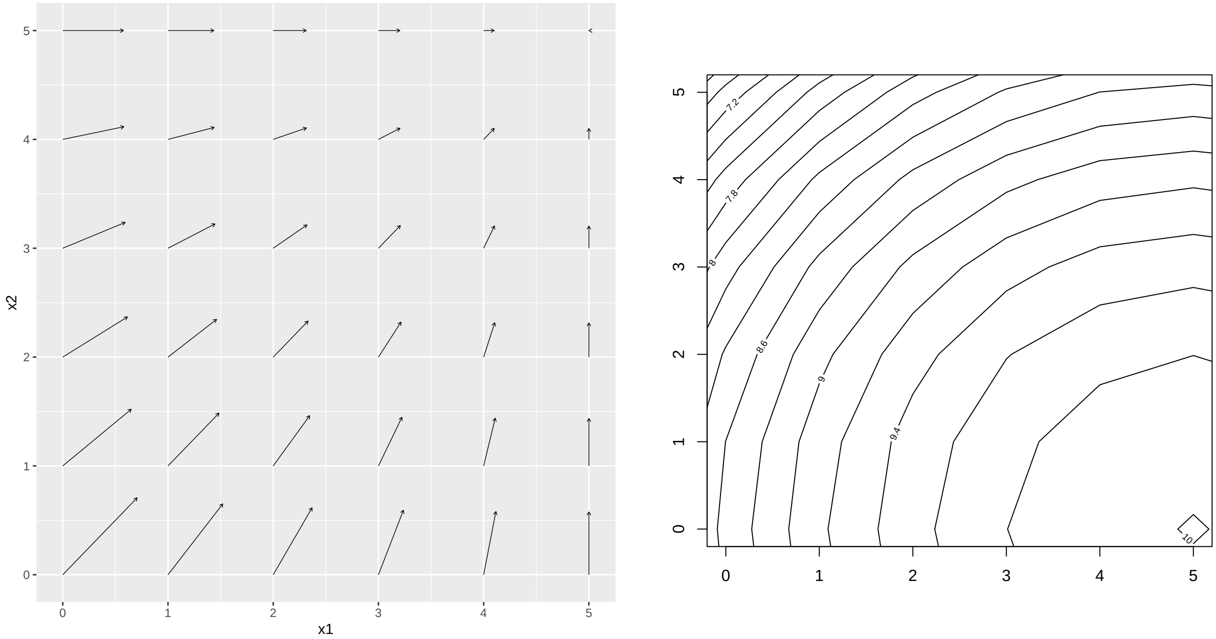

Figure 17: Slope field and contour plot for the function $g(x) = \sqrt{100 - (x_1 - 5)^2 - (x_2 - 5)^2}$

③ Normalise vectors of partial derivatives.

This is done, and a slope field showing these normalised vectors is shown in Figure 18.

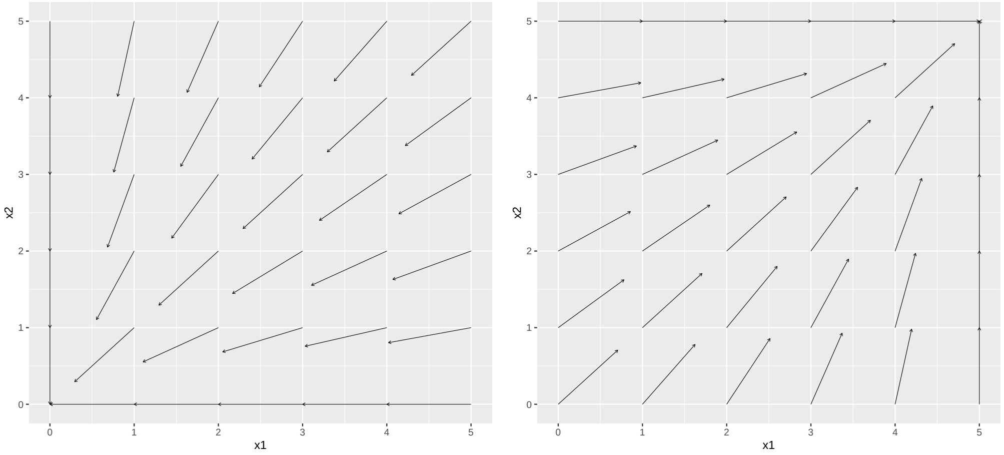

Figure 18: Normalised slope fields for the functions $f(x) = \sqrt{100 - x_1^2 - x_2^2}$ and $g(x) = \sqrt{100 - (x_1 - 5)^2 - (x_2 - 5)^2}$.

Steps ④-⑥ are performed together, and the result of Step ⑥ is discussed in Step ⑦.

④ Compute the dot product and take the absolute values.

⑤ Repeat the process over the entire parameter space.

⑥ Sum all together and divide by the number of discrete points.

⑦ Check how close this number is to 1.

For functions $f(x)$ and $g(x)$, whose contours plots are reprinted in Figure 19 below, we get a number of **0.56**.

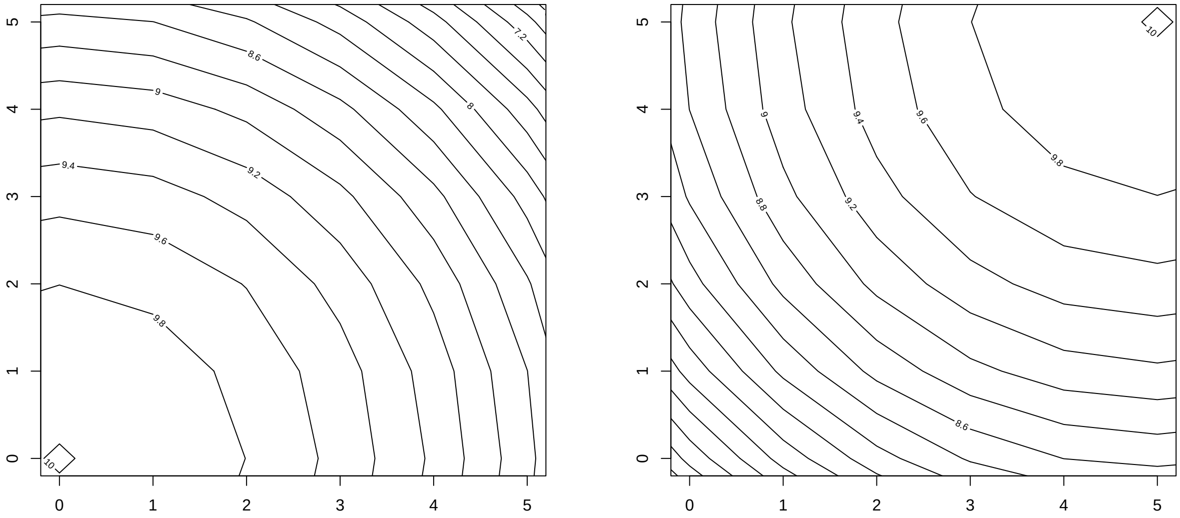

Figure 19: Contour plots for the functions $f(x) = \sqrt{100 - x_1^2 - x_2^2}$ and $g(x) = \sqrt{100 - (x_1 - 5)^2 - (x_2 - 5)^2}$.

3.3 Summary

On the next page, side-by-side contour plots for each pair of $f(x)$ and $g(x)$ from Sections 3.1 and 3.2 are presented, along with their value of the alignment measure.

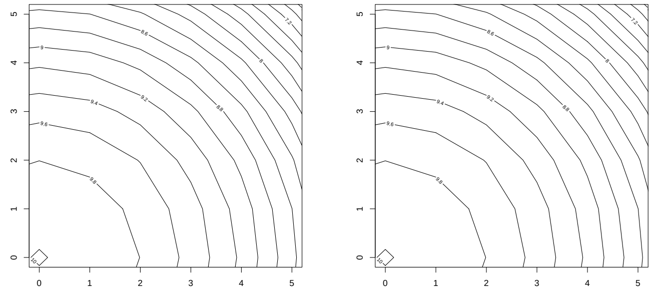

Figure 20: Alignment measure of 0.97.

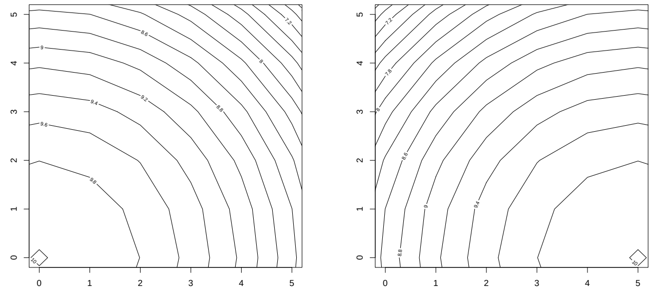

Figure 21: Alignment measure of 0.48.

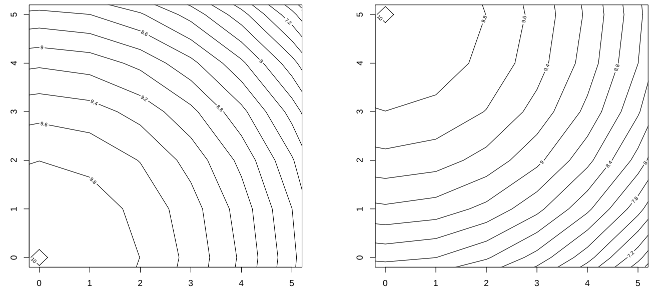

Figure 22: Alignment measure of 0.48.

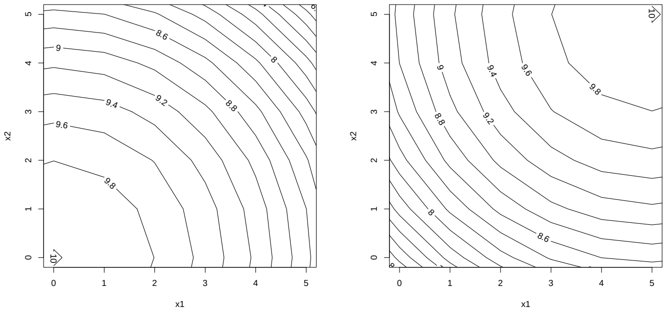

Figure 23: Alignment measure of 0.56.

Citations

Keirstead, James. 2012. "Gaussian Process Regression with r: R-Bloggers." *R*. https://www.r-bloggers.com/2012/04/gaussian-process-regression-with-r.

O'Hagan, A. 1992. "Some Bayesian Numerical Analysis." In *Bayesian Statistics 4: Proceedings of the Fourth Valencia International Meeting*. Oxford University Press. https://doi.org/10.1093/oso/9780198522669.003.0019.

Rasmussen, C. E., and C. K. I. Williams. 2006. *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press.