MA144 Methods of Mathematical Modelling 2, Assignment 2

Dyson Dyson

Question 1

Q1 (a)

The heat equation is

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$$

A solution to the heat equation is

$$u(x,t) = \frac{1}{\sqrt{2\pi t}}\,\mathrm{e}^{-x^2/2t} = \frac{1}{\sqrt{2\pi}}\,t^{-\frac{1}{2}}\,\mathrm{e}^{-x^2/2t}$$

Q1 (a) i)

The solution gives

$$\begin{aligned}
\frac{\partial u}{\partial t} &= \frac{1}{\sqrt{2\pi}}\left(t^{-\frac{1}{2}}\mathrm{e}^{-x^2/2t}\frac{x^2}{2t^2} - \frac{1}{2}t^{-\frac{3}{2}}\,\mathrm{e}^{-x^2/2t}\right) \\
&= \frac{1}{\sqrt{2\pi}}\,\mathrm{e}^{-x^2/2t}\left(t^{-\frac{1}{2}}\frac{x^2}{2}t^{-2} - \frac{1}{2}t^{-\frac{3}{2}}\right) \\
&= \frac{1}{2\sqrt{2\pi}}\,\mathrm{e}^{-x^2/2t}\left(x^2t^{-\frac{5}{2}} - t^{-\frac{3}{2}}\right) \\
&= \frac{1}{2\sqrt{2\pi}}\,\mathrm{e}^{-x^2/2t}t^{-\frac{3}{2}}\left(x^2t^{-1} - 1\right) \\
&= \frac{1}{2\sqrt{2\pi t^3}}\,\mathrm{e}^{-x^2/2t}\left(\frac{x^2}{t} - 1\right)
\end{aligned}$$

and

$$\begin{aligned}
\frac{\partial^2 u}{\partial x^2} &= \frac{\partial}{\partial x}\left(\frac{1}{\sqrt{2\pi t}}\,\mathrm{e}^{-x^2/2t}\left(\frac{-x}{t}\right)\right) \\
&= \frac{\partial}{\partial x}\left(\frac{-1}{\sqrt{2\pi t^3}}\,x\mathrm{e}^{-x^2/2t}\right) \\
&= \frac{-1}{\sqrt{2\pi t^3}}\left(\mathrm{e}^{-x^2/2t} + x\mathrm{e}^{-x^2/2t}\left(\frac{-x}{t}\right)\right) \\
&= \frac{1}{\sqrt{2\pi t^3}}\mathrm{e}^{-x^2/2t}\left(\frac{x^2}{t} - 1\right)
\end{aligned}$$

Therefore $\alpha = 2$ for this solution.

Q1 (a) ii)

$$u(x,1) = \frac{1}{\sqrt{2\pi}}\,\mathrm{e}^{-x^2/2}$$

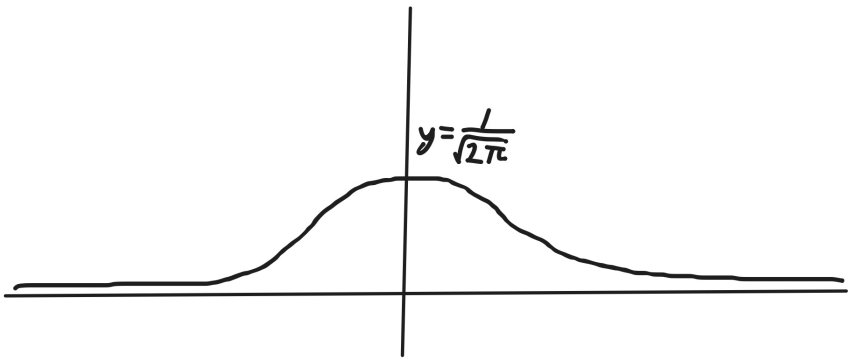

Figure 1: A plot of $y = u(x,1)$

Q1 (b)

The wave equation is

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\,\frac{\partial^2 u}{\partial t^2}$$

A solution to the wave equation is

$$u(x,t) = \sin x \cos \beta t$$

where $\beta > 0$.

Q1 (b) i)

The solution gives

$$\begin{aligned}\frac{\partial^2 u}{\partial x^2} &= \frac{\partial}{\partial x}\left(\cos x \cos \beta t\right)\\ &= -\sin x \cos \beta t\\ &= -u(x,t)\end{aligned}$$

and

$$\begin{aligned}\frac{\partial^2 u}{\partial t^2} &= \frac{\partial}{\partial t}\left(-\beta \sin x \sin \beta t\right)\\ &= -\beta^2 \sin x \cos \beta t\\ &= -\beta^2 u(x,t)\end{aligned}$$

Plugging this into the wave equation gives

$$-u(x,t) = -\frac{\beta^2}{c^2} u(x,t)$$

Therefore $\frac{\beta^2}{c^2} = 1$ so $\beta = \pm c$. We know $c > 0$ and we want $\beta > 0$, so $\beta = c$.

Q1 (b) ii)

Let $u(x,t) = f(x+ct) + g(x-ct)$. Then by the chain rule,

$$\begin{aligned}\frac{\partial^2 u}{\partial x^2} &= \frac{\partial}{\partial x}\left(f'(x+ct) + g'(x-ct)\right)\\ &= f''(x+ct) + g''(x-ct)\end{aligned}$$

and

$$\begin{aligned}\frac{\partial^2 u}{\partial t^2} &= \frac{\partial}{\partial t}\left(cf'(x+ct) - cg'(x-ct)\right)\\ &= c^2 f''(x+ct) + c^2 g''(x-ct)\end{aligned}$$

Therefore $\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 u}{\partial t^2}$, so this u satisfies the wave equation.

Question 2

$$f(x,y) = \mathrm{e}^x + y^3 - 2x - 3y$$

Q2 (a)

$$\frac{\partial f}{\partial x} = \mathrm{e}^x - 2 \qquad \frac{\partial f}{\partial y} = 3y^2 - 3$$

Critical points are where $\dfrac{\partial f}{\partial x} = \dfrac{\partial f}{\partial y} = 0$. That means $\mathrm{e}^x - 2 = 0$ so $x = \log 2$, and $3y^3 - 3 = 0$ so $y = \pm 1$. Therefore the critical points of f are at $(\log 2, 1)$ and $(\log 2, -1)$.

The Hessian matrix of f is $\begin{pmatrix} \mathrm{e}^x & 0 \\ 0 & 6y \end{pmatrix}$ and its determinant is $D = 6\mathrm{e}^x y$.

At the point $(\log 2, 1)$, $D = 12$ and $f_{xx} = 2$, so this point is a local minimum point.

At the point $(\log 2, -1)$, $D = -12$, so this point is a saddle point.

Q2 (b)

Linear approximation around $P = (0, 2)$ gives

$$\begin{aligned} f(x,y) &\approx f(0,2) + \nabla f|_{(0,2)} \cdot \begin{pmatrix} x \\ y-2 \end{pmatrix} \\ &= (1 + 8 - 0 - 6) + \left.\begin{pmatrix} \mathrm{e}^x - 2 \\ 3y^2 - 3 \end{pmatrix}\right|_{(0,2)} \cdot \begin{pmatrix} x \\ y-2 \end{pmatrix} \\ &= 3 + \begin{pmatrix} -1 \\ 9 \end{pmatrix} \cdot \begin{pmatrix} x \\ y-2 \end{pmatrix} \\ &= 3 - x + 9y - 18 \\ \therefore\ f(x,y) &\approx -15 - x + 9y \end{aligned}$$

Therefore

$$\begin{aligned} f(0.01, 1.99) &\approx -15 - 0.01 + 9 \times 1.99 \\ &= -15.01 + 9 + 8.91 \\ &= -15.01 + 17.91 \\ &= 2.9 \end{aligned}$$

For completeness, $f(0.01, 1.99)$ is actually around 2.900649, so this approximation is very good.

Q2 (c)

The direction of steepest descent at point P is $-\nabla f|_P = \begin{pmatrix} 1 \\ -9 \end{pmatrix}$.

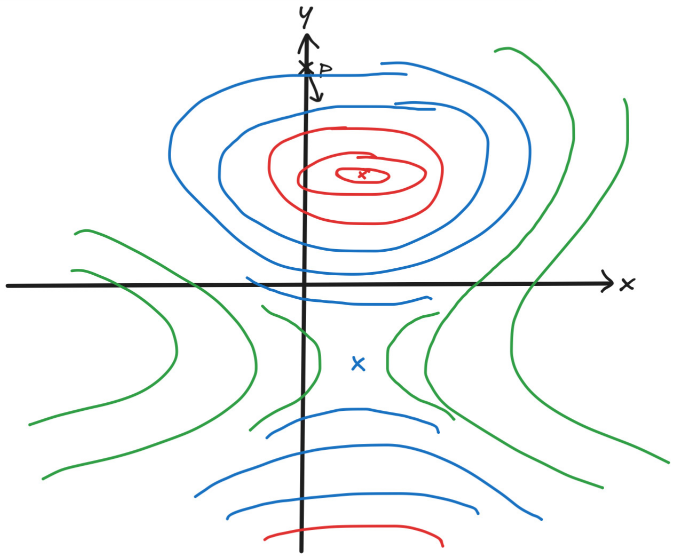

Figure 2: A contour plot of $z = f(x, y)$. Red is negative z, blue is small positive z, and green is large positive z. The red cross is the local minimum point, the blue cross is the saddle point, the black cross is P, and the black arrow is the direction of steepest descent from P.

Q2 (d)

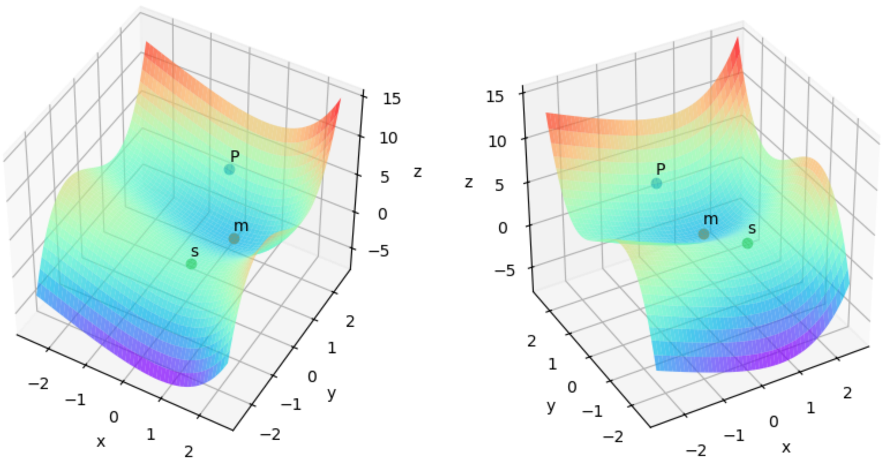

Figure 3: A 3D plot of $z = f(x, y)$ from two angles. m is the local minimum point and s is the saddle point.

```python
#!/usr/bin/env python3

from pathlib import Path
import numpy as np
import matplotlib.pyplot as plt


def f(x, y):
    return np.exp(x) + y**3 - 2 * x - 3 * y


def plot_point(ax, coords, name):
    x, y = coords
    ax.plot(x, y, f(x, y), "o")
    ax.text(x, y, f(x, y) + 1, name)


def plot_3d_from_angle(filename, *, elev, azim):
    x = np.linspace(-2.5, 2.5)
    y = np.linspace(-2.5, 2.5)

    X, Y = np.meshgrid(x, y)

    fig = plt.figure()
    ax = fig.add_subplot(projection="3d")
    ax.set_box_aspect((1, 1, 1))
    ax.view_init(elev=elev, azim=azim)

    ax.plot_surface(X, Y, f(X, Y), alpha=0.7, cmap="rainbow")
    ax.set_xlabel("x")
    ax.set_ylabel("y")
    ax.set_zlabel("z")

    # Plot point P
    plot_point(ax, (0, 2), "P")

    # Plot critical points
    plot_point(ax, (np.log(2), 1), "m")
    plot_point(ax, (np.log(2), -1), "s")

    plt.savefig(
        Path(__file__).parent.parent / "imgs" / f"{filename}.png",
        bbox_inches="tight",
        pad_inches=0.2,
    )
    plt.clf()


def main() -> None:
    plot_3d_from_angle("Q2d-1", elev=45, azim=-60)
    plot_3d_from_angle("Q2d-2", elev=35, azim=-120)


if __name__ == "__main__":
    main()
```

Figure 4: The code used to generate Figure 3. The code can also be found on GitHub.