This problem set involves some programming; you may use whatever you want for this, but I strongly recommend you use either `Matlab` (or something similar, such as `R`) or a compiled programming language (e.g., `C++`) for this purpose. If you have never done any programming, please contact me and we will figure something out.

Q. 1. Consider the stochastic differential equations

$$dX_t^r = \sin(X_t^r)\,dW_t^1 + \cos(X_t^r)\,dW_t^2, \quad X_0^r = r, \qquad dY_t^r = dW_t^1, \quad Y_0^r = r,$$

where $r \in \mathbb{R}$ is non-random and (W_t^1, W_t^2) is a two-dimensional Wiener process.

1. Show that X_t^r has the same law as Y_t^r for every fixed time t.

   [**Hint:** investigate the Kolmogorov backward equations for X_t^r and Y_t^r.]

2. Show that X_t^r has independent increments. Together with the previous part, this implies that $\{X_t^r\}$ is a one-dimensional Brownian motion started at r.

   [**Hint:** show that $\mathbb{E}(f(X_t^r - X_s^r)|\mathcal{F}_s) = \mathbb{E}(f(X_t^r - z)|\mathcal{F}_s)|_{z=X_s^r} \equiv g(X_s^r)$ is constant, i.e., the function $g(x)$ is independent of x (you do not need to prove the first equality; it follows as in the proof of lemma 3.1.9). Then show why this implies $\mathbb{E}(f(X_t^r - X_s^r)Z) = \mathbb{E}(f(X_t^r - X_s^r))\,\mathbb{E}(Z)$ for any $\mathcal{F}_s$-measurable Z.]

X_t^r is thus a Brownian motion started at r—what more can be said? Surprisingly, X_t^r and Y_t^r behave very differently if we consider multiple initial points $r_1, \ldots, r_n$ simultaneously, *but driven by the same noise*. In other words, we are interested in

$$Y_t = (Y_t^{r_1}, \ldots, Y_t^{r_n}) = (r_1 + W_t^1, \ldots, r_n + W_t^1), \qquad X_t = (X_t^{r_1}, \ldots, X_t^{r_n}),$$

where the latter is the solution of the n-dimensional SDE every component of which satisfies the equation for X_t^r above.

3. Use the Euler-Maruyama method to compute several sample paths of X_t and of Y_t in the interval $t \in [0, 10]$, with $(r_1, \ldots, r_n) = (-3, -2.5, -2 \ldots, 3)$ and with step size $\Delta t = .001$. Qualitatively, what do you see?

Apparently the SDEs for X_t^r and Y_t^r are qualitatively different, despite that for every initial condition their solutions have precisely the same law! These SDEs generate the same Markov process, but a different *flow* $r \mapsto X_t^r$, $r \mapsto Y_t^r$. Stochastic flows are important in the theory of random dynamical systems (they can be used to define Lyapunov exponents, etc.), and have applications, e.g., in the modelling of ocean currents.

Q. 2. We are going to investigate the inverted pendulum of example 6.6.5 in the lecture notes, but with a different cost functional. Recall that we set

$$d\theta_t^u = c_1 \sin(\theta_t^u)\, dt - c_2 \cos(\theta_t^u)\, u_t\, dt + \sigma\, dW_t.$$

As the coefficients of this equation are periodic in θ, we may interpret its solution modulo 2π (i.e., θ_t^u evolves on the circle, which is of course the intention).

Our goal is to keep θ_t^u as close to the up position $\theta = 0$ as possible on some reasonable time scale. We will thus investigate the discounted cost

$$J_\lambda[u] = \mathbb{E}\left[\int_0^\infty e^{-\lambda s}\{p\,(u_s)^2 + q\,(1 - \cos(\theta_s^u))\}\, ds\right].$$

This problem does not lend itself to analytic solution, so we approach it numerically.

1. Starting from the appropriate Bellman equation, develop a Markov chain approximation to the control problem of minimizing $J_\lambda[u]$ following the finite-difference approach of section 6.6 in the lecture notes. Take the fact that θ_t^u evolves on the circle into account to introduce appropriate boundary conditions.

   [**Hint:** it is helpful to realize what the discrete dynamic programming equation for a discounted cost looks like. If x_n^α is a controlled Markov chain with transition probabilies $P_{i,j}^\alpha$ from state i to state j under the control α, and

   $$K_\varrho[u] = \mathbb{E}\left[\sum_{n=0}^\infty \varrho^n\, w(x_n^u, u_{n+1})\right], \qquad 0 < \varrho < 1,$$

   then the value function satisfies $V(i) = \min_{\alpha\in\mathbb{U}}\{\varrho \sum_j P_{i,j}^\alpha V(j) + w(i,\alpha)\}$. You will prove a verification theorem for such a setting in part 2.]

2. To which discrete optimal control problem does your numerical method correspond? Prove an analog of proposition 6.6.2 for this case.

3. Using the Jacobi iteration method, implement the numerical scheme you developed, and plot the optimal control and the value function.

   You can try, for example, $c_1 = c_2 = \sigma = .5$, $p = q = 1$, $\lambda = .1$; a grid which divides $[0, 2\pi[$ into 100 points; and 500 iterations of the Jacobi method (but play around with the parameters and see what happens, if you are curious!)