QNT 402 Quantum Information
Homework 7

1. [35] *A simple model test.* In this problem we'll generate some fake data and then try to fit it to determine if modeling it as linear or quadratic function is more reasonable.
   a.) [5] First generate some fake data using the Python code:
   ```
   slope = 1
   offset = 0.25
   time = np.linspace(0,10,10)
   data = np.random.normal(slope*time + offset,1)
   sigma = 1*np.random.normal(np.ones(len(data)),0.01)
   ```
   and plot it with the pyplot errorbar() function. This is obviously linear data but it's pretty noisy as you'll see.
   b.) [15] Now use curve_fit to fit a linear function ($bx + a$) and a quadratic function ($cx^2 + bx + a$) and add the plot of the best fits to the plot with the data on it.
   c.) [15] Now calculate the χ^2/ν of each of these and try to determine which is a more likely model.

2. [25] *Variance vs. Slope detection: a more accurate comparison.* In class, we found that the minimum sensitivity for slope detection was:

$$\Phi_{min}^{slope} = \frac{\sigma_p}{\gamma\, t}$$

where, assuming the qubit state detection is given by a binomial distribution, $\sigma_p = \frac{\sigma_{N1}}{N} = \sqrt{\frac{p_1(1-p_1)}{N}}$. We said for $p_1 \approx \frac{1}{2}$ this leads to the standard quantum limit of:

$$\Phi_{min}^{slope} = \frac{1}{2\, \gamma\, t\, \sqrt{N}}$$

Meanwhile for variance detection we found

$$\Phi_{min}^{variance} = \frac{\sqrt{\sigma_p}}{\gamma\, t}$$

We then said for detection at $p \approx 0$ we expect $\sigma_p = 0$ and therefore variance detection is not limited by quantum projection noise.

However, the approximation that $p_1 \approx \frac{1}{2}$ for slope detection and $p_1 \approx 0$ for variance detection are not quite true. Obviously if we are detecting a signal there is some change in p_1, so we have for slope detection $p_1 \approx \frac{1}{2} + \delta p_1$ and $p_1 \approx \delta p_1$. Therefore, the uncertainty on σ_p is a bit different than we estimated above.

   a.) [20] Work out the expressions for Φ_{min} both slope and variance including the corrections to p_1 above. Make sure and write δp_1 in terms of the sensitivity γ, the field strength Φ, and the phase accumulation time t.
   b.) [5] What is the ratio of $\Phi_{min}^{slope}/\Phi_{min}^{variance}$? Ignoring technical noise (which you can't do in the real world!), when is variance detection better than slope detection?

3. [50] *Alright, alright, alright, let's work with some quantum data.* In this problem we will generate data from a quantum sensing experiment. We know that for a Ramsey sequence (we worked it out last quarter) that for a qubit starting in $|0\rangle$ the probability of finding the qubit in state 1 is given as:

$$P_1 = 1 - \frac{\Omega^2}{\Omega'^4}\left(\Omega' \cos\left(\frac{\delta t_w}{2}\right)\sin\left(\Omega' t_p\right) - 2\delta \sin\left(\frac{\delta t_w}{2}\right)\sin^2\left(\frac{\Omega' t_p}{2}\right)\right)^2$$

This expression is for two $\pi/2$ pulses of length t_p separated by a free evolution time of t_w.

a.) [20] Write a Python function that generates realistic data for N repetitions of such a sequence. This function will output a value for P_1 that comes from a binomial distribution described by the true value of P_1 and the number of trials N. Use this function to make a plot of the "measured" values of P_1 versus t_w for the a binomial the parameters $\Omega = 2\pi$ and $\delta = \frac{\pi}{4}$ for N = 1,10,100,1000.

b.) [20] Now simulate the signal for slope detection. Find the wait time for the first $P_1 = \frac{1}{2}$ point (i.e. the one near $t_w = 2$) and plot P_1 vs Φ for $-1 \le \Phi \le 1$. Assume $\gamma = 1$. Make the plots for N = 1,10,100,1000. Also plot the expected signal we derived in class.

c.) [10] Take the plot you made for N = 100 and add the expected signal plot to it as well as the expected signal plus σ_p and the expected signal minus σ_p. Is it what you expect?

4. [60] *One last dance with the QHO.* Here, we're going to examine the effect of squeezing on a QHO.

a.) [10]Define a Python function that that takes time, t, as its input and returns the time evolution operator a time t for the Hamiltonian $H_o = \hbar\omega\left(\hat{a}^\dagger\hat{a} + \frac{1}{2}\right)$. Use that function to calculate $|\alpha(t)\rangle = \hat{U}(t)|\alpha\rangle$ for various times during the oscillation period of the oscillator. Plot the expectation value of position and momentum in the (x,p) plane for $\alpha = 1$ – make sure to make your basis is big enough for this size coherent state. Choose $\omega = 2\pi$ and $m = 1$. Here, you'll likely find QuTip's displace() function handy.

b.) [10] Calculate σ_x and σ_p at each of the times you chose in part (a) and plot σ_x, σ_p, and $\sigma_x\sigma_p$ versus time.

c.) [15] QuTip has a function called squeeze() that implements the squeeze operator we discussed in class. Create a squeezed coherent state by first applying the squeeze operator (with squeeze parameter r = 1; fyi, QuTip uses z for the parameter r from class) and then applying the displacement operator used in part (a). Now, remake the plots of part (a) and part(b). What do you notice?

d.) [20] The squeeze operator is a bit of shortcut, so let's make sure we know how to implement it on real hardware. Add a drive to QHO of the form $V = \beta_1\, x\sin(2\omega t) + \beta_2\, x^2 \sin(2\omega t)$ use e.g. sesolve to find the evolution. Choose $\beta_1 = \beta_2 = 1$. Make the same plots you made in part (a) and part (b). Is this squeezing?

e.) [5] What are the two terms in the potential V? Try setting e.g. $\beta_1 = 0$ and describe what you see.