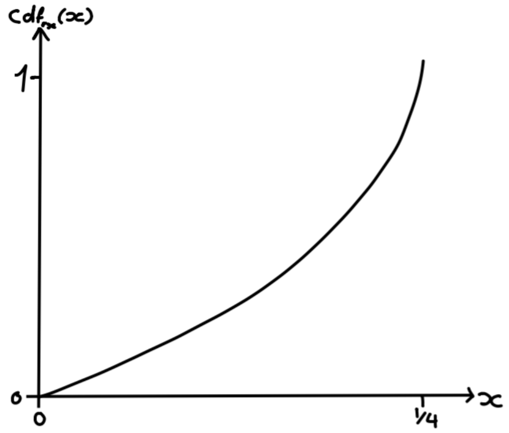
1 Example Sheet 3

Before all code I assume all required data has been cleaned, arguments are named as described in the question and the following imports have been made:

```
import numpy as np
import scipy.stats as stats
from sklearn.linear_model import LinearRegression
```

1. Sketch the cumulative distribution function and calculate the density function for this continuous random variable:

```
def rx():
        u = random.random()
        return u * (1 - u)
```



$$
\begin{aligned}
P(rx < x) &= P(u(1-u) < x) \\
&= P(u^2 - u + x > 0) \\
&= P\left(u < \frac{1-\sqrt{1-4x}}{2}\right) + P\left(u > \frac{1+\sqrt{1-4x}}{2}\right) \\
&= \frac{1-\sqrt{1-4x}}{2} + 1 - \frac{1+\sqrt{1-4x}}{2} \\
&= 1 - \sqrt{1-4x}
\end{aligned}
$$

This is the cumulative distribution function. We can obtain the density function by differentating this:

$$
\begin{aligned}
pdf_{rx}(x) &= \frac{\partial}{\partial x}\left(1 - \sqrt{1-4x}\right) \\
&= \frac{2}{\sqrt{1-4x}}
\end{aligned}
$$

2. We are given a dataset $x_1, \dots, x_n$ which we believe is drawn from a $\mathcal{N}(\mu, \sigma^2)$ where μ and σ are unknown.

   (a) Find the maximum likelihood estimators for $\hat{\mu}$ and $\hat{\sigma}$.

```
x = [...]
sigma = np.std(x)
mu = np.mean(x)
```

(b) Find a 95% confidence interval for $\hat{\sigma}$ using parametric resampling.

```
x = [...]
n = ...
stds = np.std(stats.norm.rvs(np.mean(x), np.std(x), size=(n, len(x))), axis=1)
np.quantile(stds, [0.025, 0.975])
```

(c) Repeat, but using non-parametric resampling.

```
x = [...]
n = ...
stds = np.std(np.random.choice(x, size=(n, len(x))), axis=1)
np.quantile(stds, [0.025, 0.975])
```

3. The number of unsolved murders in Kembleford over three successive years was 3, 1, 5. The police chief was then replaced and the numbers over the following two years were 2, 3. We know from general policing knowledge that the number of unsolved murders in a given year follows the Poisson distribution. Model the numbers as Poisson(μ) under the old chief and Poisson(ν) under the new chief.

(a) Report a 95% confidence interval for $\hat{\nu} - \hat{\mu}$ using parametric sampling.

The maximum likelihood estimator for the parameter for a Poisson distribution is the mean of the distribution.

```
old = [3, 1, 5]
new = [2, 3]
n = ...
mu_hat = np.mean(old)
nu_hat = np.mean(new)
mu_resamp = np.mean(stats.poisson.rvs(mu_hat, size=(n, 3)), axis=1)
nu_resamp = np.mean(stats.poisson.rvs(nu_hat, size=(n, 2)), axis=1)
difs = nu_resamp - mu_resamp
np.quantile(difs, [0.025, 0.975])
```

With $n = 10000$, the confidence interval obtained is $\hat{\nu} - \hat{\mu} \in [-3.3, 2.5]$.

(b) Conduct a hypothesis test of the hypothesis $\mu = \nu$, using parametric sampling and using the test statistic $\nu - \mu$. Explain your choice between a one-sided and a two-sided test.

We should use a two-sided test – if $\nu - \mu$ is very small or very large then we will reject the null hypothesis H_0.

Consider the null hypothesis $H_0 : \mu = \nu$. We can use this assumption to form a 95% confidence interval for $\hat{\nu} - \hat{\mu}$. If the observed $\hat{\nu} - \hat{\mu}$ lies outside this interval then we can reject the null hypothesis H_0.

```
old = [3, 1, 5]
new = [2, 3]
n = ...
mu = np.mean(np.concatenate((old, new)))
old_resamp = stats.poisson.rvs(mu, size=(n, 3))
new_resamp = stats.poisson.rvs(mu, size=(n, 2))
statistic = np.mean(new_resamp, axis=1) - np.mean(old_resamp, axis=1)
np.quantile(statistic, [0.025, 0.975])
```

Using $n = 100000$ the 95% confidence interval obtained for $\hat{\nu} - \hat{\mu}$ was $[-3, 3]$.

Since the observed test metric, $\hat{\nu} - \hat{\mu}$ was $-\frac{1}{6}$; which lies in the 95% confidence interval, we do not have sufficient evidence to reject the null hypothesis that $\mu = \nu$.

(c) Explain carefully the difference in sampling methods between parts (a) and (b).

In part (a), we did not assume the distributions were the same. Therefore we calculated two different means and resampled from different distributions. While in the second case, we were creating a confidence interval for $\hat{\nu} - \hat{\mu}$ under the assumption that both data were drawn from the same distribution. Therefore we calculated a mean for the data together and sampled from that.

4. In section 2.2 we considered a climate model in which temperatures increase linearly. The probabilistic version of the model is

$$\mathbf{temp} \sim \alpha + \beta_1 \sin(2\pi\mathbf{t}) + \beta_2 \cos(2\pi\mathbf{t}) + \gamma(\mathbf{t} - 2000) + \mathcal{N}(0, \sigma^2)$$

Find a 95% confidence interval for $\hat{\gamma}$, the maximum likelihood estimator for the rate of temperature increase.

```python
n = ...
features = np.column_stack([
          np.ones_like(climate.t),
          np.sin(climate.t),
          np.cos(climate.t),
          climate.t
])
model = LinearRegression(fit_intercept=False).fit(features, climate.temp)
model_predict = model.predict(features)
std = np.std(model_predict - climate.temp)
gammas_resamp = np.array([
          LinearRegression(fit_intercept=False).fit(features,
                    stats.norm.rvs(model_predict, std)).coef_[3]
                              for _ in tqdm(range(n))
])
np.quantile(gammas_resamp, [0.025, 0.975])
```

Using the data from Cambridge, the 95% confidence interval obtained is $\hat{\gamma} \in [0.00857158, 0.04676771]$.

5. I have defined a function for computing the fitted temperature at an arbitrary future timepoint

```python
def pred(t):
          return α̂ + β̂1sin(2πt) + β̂2cos(2πt) + γ̂(t - 2000)
```

Modify the code to also return a 95% confidence interval:

```python
def pred(t):
          return stats.norm.ppf([0.025, 0.975],
                    α̂ + β̂1sin(2πt) + β̂2cos(2πt) + γ̂(t - 2000), σ̂)
```

6. To allow for non-linear temperature increase, example sheet 1 suggested a model with a step function,

$$\mathbf{temp} \sim \beta_1 \sin(2\pi\mathbf{t}) + \beta_2 \cos(2\pi\mathbf{t}) + \gamma_{\text{decade}} + \mathcal{N}(0, \sigma^2)$$

Find a 95% confidence interval for $\hat{\gamma}_{\text{2010s}} - \hat{\gamma}_{\text{1980s}}$. Conduct a hypothesis test of whether $\gamma_{\text{1980s}} = \gamma_{\text{2010s}}$.

```
n = ...
features = np.column_stack([
        np.sin(climate.t),
        np.cos(climate.t),
        *[
                climate.t // 10 == i for i in range(195, 203)
        ]
])
model = LinearRegression(fit_intercept=False).fit(features, climate.temp)
std = np.std(model.predict(features) - climate.temp)
resamp = np.array([
        LinearRegression(fit_intercept=False).fit(features,
                stats.norm.rvs(climate.temp, std)).coef_
                        for _ in tqdm(range(n))
])
np.quantile(resamp[:, 9] - resamp[:, 6], [0.025, 0.975])
```

Let the null hypothesis H_0 be $\gamma_{1980\text{s}} = \gamma_{2010\text{s}}$. We can perform a hypothesis test by finding a 95% confidence interval for the difference between $\gamma_{1980\text{s}}$ and $\gamma_{2010\text{s}}$ given the observed data. If this interval contains 0 then we will not be able to reject H_0 based on the observed data.

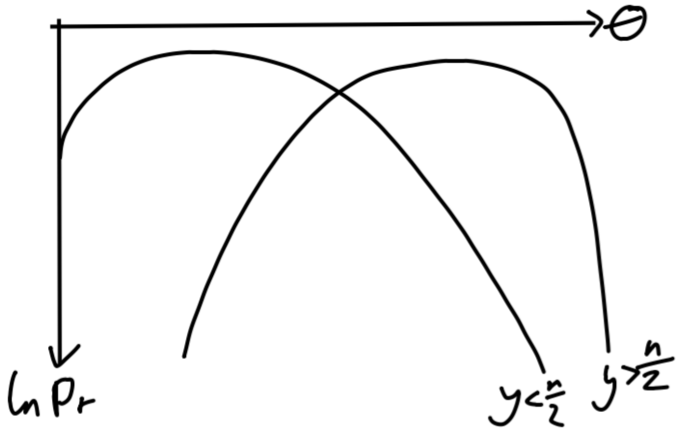
With $n = 10000$, the 95% confidence interval obtained using the data from Cambridge was $[-1.86936438, 2.87835352]$. Since 0 is in this interval, we do not have sufficient evidence to reject the null hypothesis H_0.

7. I toss a coin n times and get the answers $x_1, \dots, x_n$. My model is that each toss is $X_i \sim B(1, \theta)$ and I wish to test the null hypothesis H_0 that $\theta \geq \frac{1}{2}$.

   (a) Find an expression for $\Pr(x_1, \dots, x_n; \theta)$. Give your expression as a function of $y = \sum_i x_i$.

$$\Pr(x_1, \dots, x_n) = \binom{n}{y} \theta^y (1-\theta)^{n-y}$$

   (b) Sketch $\ln \Pr(x_1, \dots, x_n; \theta)$ as a function of θ, for two cases: $y < \frac{n}{2}$ and $y > \frac{n}{2}$.



   (c) Assuming H_0 is true, what is the maximum likelihood estimator for θ?

$$\hat{\theta} = \max\left(\frac{y}{n}, \frac{1}{2}\right)$$

(d) Let the test statistic be y. What is the distribution of this test statistic when θ is equal to your value from part (c).

$$Y \sim B(n, \hat{\theta})$$

(e) Explain why a one-sided hypothesis test is appropriate. Give an expression for the p-value of the test.

We are only interested in the probability that $\theta < \frac{1}{2}$. We are not interested in the possibility that it is greater than $\frac{1}{2}$ and therefore should not be convinced by a very large value for y – however this is what a two-sided test would do.

$$p = \sum_{i=0}^{y} \binom{n}{i} \frac{1}{2}^{i} \cdot \frac{1}{2}^{n-i}$$

8. Your attempts at a task succeed with probability θ and fail with probability $1 - \theta$. How long an unbroken list of failures does it take for you to reject "$\theta \geq \frac{1}{2}$" at p-value 5%?

Assuming the only data that we have is this list of failures, a list of failures of length $\lceil -\log_2 0.05 \rceil = 5$ would be sufficient to convince us with 95% confidence that the probability of success was $< \frac{1}{2}$

2 Supplementary questions

9. A point lightsource at coordinates (0, 1) sends out a ray of light at an angle Θ chosen uniformly in $\left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$. Let X be the point where the ray intersects the horizontal line through the origin. What is the density of X?

$$\begin{aligned} P(X < x) &= P(\tan(u) < x) \\ &= P(u < \arctan(x)) \\ &= \frac{\frac{\pi}{2} + \arctan(x)}{\pi} \\ &= c \end{aligned}$$

We can differentiate this to find the probability density function for x:

$$\begin{aligned} \text{pdf}(x) &= \frac{\partial}{\partial x}\left(\frac{\pi + 2\arctan(x)}{2\pi}\right) \\ &= \frac{1}{\pi(1 + x^2)} \end{aligned}$$

10. We are given a dataset $x_1, \ldots, x_n$ which we believe is drawn from $\mathcal{U}[0, \theta]$ where θ is unknown. Recall from example sheet 1 that the maximum likelihood estimator is $\hat{\theta} = \max_i x_i$. Find a 95% confidence interval for $\hat{\theta}$, both using parametric resampling and using non-parametric resampling.

Parametric Resampling:

```
num_tests = ...
rands = np.random.random((num_tests, x.size))
max_gen = np.max(rands, axis=1)
ratio = 1 / max_gen
np.max(x) * np.quantile(ratio, [0, 0.95])
```

Non-parametric Resampling:

```
largest = np.max(x)
gen_data = np.random.choice(x, size=(num_tests, x.size))
max_gen = np.max(gen_data, axis=1)
ratio = np.max(x) / max_gen
np.max(x) * np.quantile(ratio, [0, 0.95])
```

11. I implement the two resamplers from question 10. To test them, I generate 1000 values from $\mathcal{U}[0, \theta]$ with $\theta = 2$ and find a 95% confidence interval for $\hat{\theta}$. I repeat this 20 times. Not once does my confidence interval include the true value, $\theta = 2$ for either resampler. Explain.

    I did not observe behaviour like this:

    - For the parametric resampler, I observed 95% of confidence intervals containing the true maxima – this result occurred for all 95% confidence intervals ie [0%, 95%], [2.5%, 97.5%], [5%, 100%].
    - For the non-parametric resampler, I observed between 80% and 90% of confidence intervals containing the correct value with 1000 attempts (depending on which confidence interval I chose).

    I believe the answer Damon wants is along the lines of: “by definition this question is asking us to extrapolate”. We are not fitting (and cannot fit) any model which is good at extrapolating. Therefore the question is inherently unsuited to frequentist approaches and even a well-fit model will not perform well when we are forced to extrapolate.

12. Test the hypothesis that temperatures in Cambridge have not been changing, using a non-parametric test.

    I will assume that temperatures in each month are sampled from some distribution. Therefore I will only shuffle the months. I will non-parametrically resample the temperature for a given day in a month from the set of temperatures which happened in that month. I will then use the MLE for γ as the readout statistic and find a confidence interval on that.

    The code is implemented below:

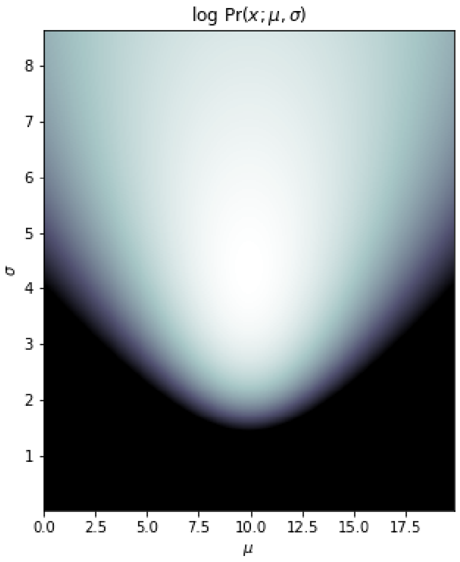
```
gammas = []

for tests in tqdm(range(25000)):
    temp = np.copy(climate.temp)
    for i in range(12):
        temp[climate.mm == i] = np.random.choice(
                temp[climate.mm == i], size=np.count_nonzero(climate.mm == i))
    gammas.append(LinearRegression(fit_intercept=False).fit(features, temp).coef_[3])
```

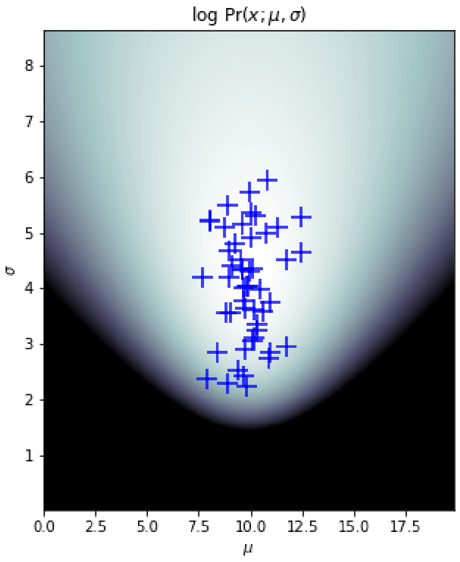
The 95% confidence interval fora two tailed test is $[-0.000549, 0.010008]$. The observed $\hat{\gamma}$ is 0.029218927735124786 – which is outside the range. Therefore, we can conclude with 95% confidence that the temperature in Cambridge is changing.

13. We have a dataset $x_1, x_2, \ldots, x_n$ and we wish to model it as $\mathcal{N}(\mu, \sigma^2)$ where μ and σ are unknown. How different are Bayesian and frequentist confidence intervals for the mean? To be concrete, lets work with the first 10 values for `temp` in the climate dataset.

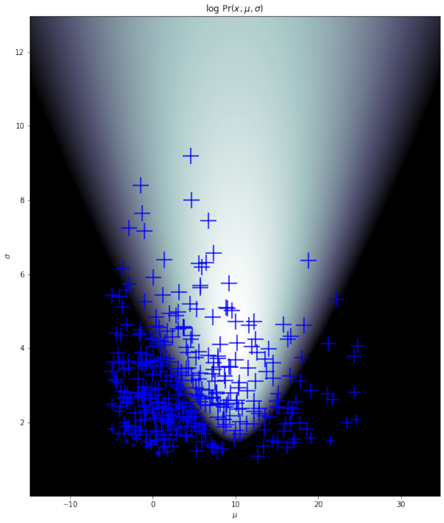
    (a) Plot the log likelihood function $\ln \Pr(x_1, \ldots, x_n | \mu, \sigma)$ as a function of μ and σ.



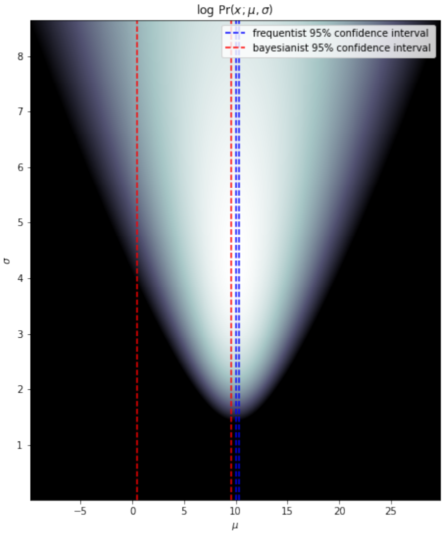
(b) Using frequentist resampling, generate 50 resampled datasets, find the maximum likelihood estimators $\hat{\mu}$ and $\hat{\sigma}$ for each and show these 50 points on your plot.



(c) Using computational Bayesian methods, with priors $\mu \sim \mathcal{N}(0, 10^2)$ and $\sigma \sim \Gamma(k = 2, \theta = 1)$ (where k) and θ are as in the numpy documentation), sample 500 pairs from the prior distribution and show them on your plot. Then compute the posterior weights of these sampled pairs and show the weigthed pairs on your plot by setting the size of the plot marker in proportion to weight.

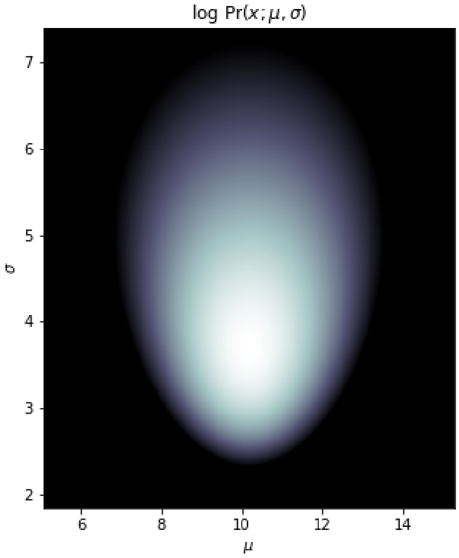


(d) Find the 95% confidence interval (for $\hat{\mu}$ in the frequentist case and for $(\mu|\text{data})$ in the Bayesianist case) and show them on your plot.

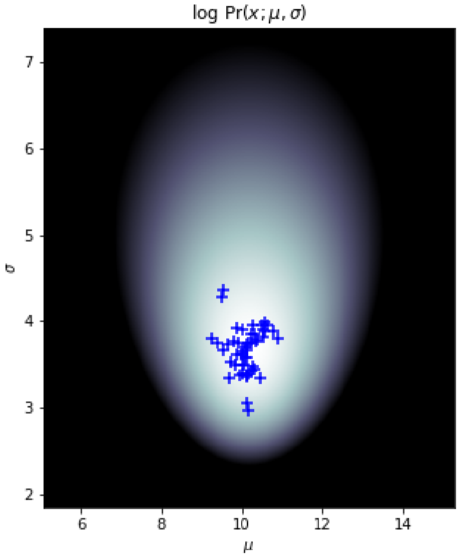


(e) Repeat the exercise, using the first 100 values from the climate dataset.

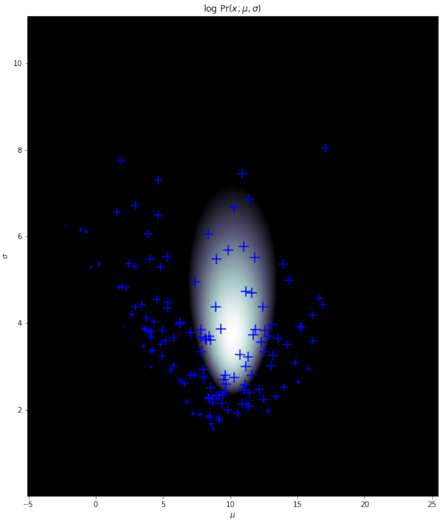
A plot of the log likelihood function of the first 100 data values versus μ, σ:



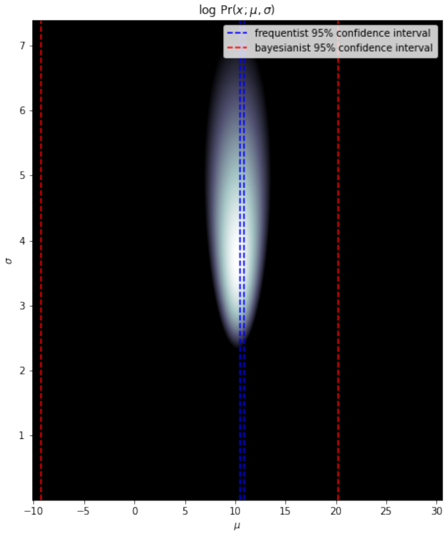
A scatter diagram of the observed means and stanard deviations of parametrically resampled datasets:



A scatter diagram of the bayesian samples with cross size scaled by log probability:



Bayesian and frequentist confidence intervals:



14. In hypothesis testing, what p-value would you expect if H_0 is true?

    If H_0 is true, I would expect $p \sim \mathcal{U}[0, 1]$.

3 2020 Paper 6 Question 8

This table shows a summary of temperature readings from the Cambridge weather station, comparing June, July and August in the 1970s to the 2010s. It shows the number of months in which the average maximum daily temperature was low (< 15.5°C), high (> 18°C) or medium. We wish to establish whether there is a significant difference between the two rows.

https://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2020p6q8.pdf

	low	med	high
1970s	10	18	2
2010s	5	14	11

Suppose that readings are independent from month to month. let $p_{d,k}$ be the probability that a month's reading falls into bin $k \in \{\text{low}, \text{med}, \text{high}\}$ in decade $d \in \{1970s, 2010s\}$. The $p_{d,k}$ are unknown parameters.

(a) Give expressions for the maximum likelihood estimates $\hat{p}_{d,k}$. In your answer, you should state what is being maximised over what variables.

$$\hat{p}_{d,k} = \frac{n_{d,k}}{n_d}$$

Where:

- $p_{d,k}$ is the probability of the months reading falling into the bin k.
- $n_{d,k}$ is the number of months in decade d which fell into bin k.
- n_d is the number of months in decade d.

(b) Let the null hypothesis H_0 be that the probabilities are the same in the 1970s as in teh 2010s; call these common probabilities q_k. Give expressions for the maximum likelihood estimates $\hat{q}_k$ under H_0.

$$\hat{q}_k = \frac{n_k}{n}$$

Where:

- n_k is the number of months in the 1970s and 2010s where the reading fell into bin k.
- n is the number of months in the 1970s and 2010s.

(c) We wish to test H_0 using the test statistic

$$t = \sum_{d,k} \frac{\left(\hat{p}_{d,k} - \hat{q}_k\right)^2}{\hat{q}_k}$$

(i) Explain briefly what is meant by *parametric resampling*. Explain how to compute the distribution we'd expect to see for t under H_0. Give pseudocode.

Parametric resampling is the process of approximating the distribution for a readout statistic by fitting a model to the data and then resampling the data from this model and generating the readout statistic for each of those data. We can then visualise this distribution of statistics either by creating confidence intervals or by making a histogram.

```
$\hat{\theta}$ = calculate_statistic(data)
resampled = fitted_model.rvs(size=(n, *data.shape))
statistic = calculate_statistic_vectorised(resampled)
low, high = quantile(statistic, [$p_{lo}$, $p_{hi}$])
plt.hist(statistic)
```

(ii) Explain what is meant by a one-sided test versus a two-sided test. Which should we use in this case?

A two-sided test can be used to test if a statistic is *different* to a particular value; while one-sided tests are used to test whether a statistic is greater than a value; or a different one-sided test can be used to test whether a statistic is less than a value.

We should use a two-sided test when we wish to test for significant evidence that a statistic is *not equal to* a certain value. We use a one-sided test to test when we have a belief about whether the statistic is greater than or less than that value – for example testing if a probability is $> \frac{1}{2}$.

(iii) Give pseudocode to compute the p-value of this test.

In this case, we are testing the assumption H_0 and therefore we should perform a two-sided hypothesis test.

```
# calculate maximum likelihood estimators from observed data
$\hat{p}$, $\hat{q}$ = calculate_mles(data)
# generate n new resampled data
resampled = np.choice(['lo','med','hi'], p=$\hat{q}$, size=(n, *data.shape))
# calculate the statistic we are interested in for each
# of these resampled distributions
resampled_t = [t(calculate_mles(resampled_i)) for resampled_i in resampled]
# work out a two-sided 95% confidence interval
lo, hi = np.quantile(resampled_t, [0.025, 0.975])
# if false we can reject $H_0$
lo < t($\hat{p}$, hatq) < hi
```

(d) What are some advantages and disadvantages of this count-based test, compared to a test based on linear regression?

Advantages:

- Results are easier to interpret
- Making good statistics is easier than a linear regression based test
- Can be less computationally expensive

Disadvantages:

- Does not fully use data
- Counts may be adversarially chosen, leading to invalid conclusions
- Ignores extreme values