Exam GEO4300

Candidate number 15413

Exercise 1

A

The expected value is the weighted (by probability) average of all possible values:

$$E(x) = -1 \cdot \frac{1}{3} + 3 \cdot \frac{1}{2} + 4 \cdot \frac{1}{6} = \frac{11}{6}$$

B

The variance is:

$$V(x) = \frac{\sum (X_i - \mu)^2}{n} = \frac{\frac{1}{3}(-1 - \frac{11}{6})^2 + \frac{1}{2}(3 - \frac{11}{6})^2 + \frac{1}{6}(4 - \frac{11}{6})^2}{1} = 4.139$$

C

The mode is the most commonly occuring observation. In this case that is 3, since it has the highest probability of occuring.

D

$$C_v = \frac{\sqrt{V(x)}}{E(x)} = 1.11$$

In [1]:
```
import scipy as sc

#We can confirm these results with

n = 1e5
a = sc.zeros(int(n))
f1 = int(1/3*n)
f2 = int(1/2*n)
f3 = int(1/6*n)
a[:f1] = -1
a[f1:(f1+f2)] = 3
a[-f3:] = 4

print(f"mean: {sc.mean(a)}")
print(f"var: {sc.var(a)}")
print(f"cv: {sc.sqrt(sc.var(a))/sc.mean(a)}")
```

```
mean: 1.83331
var: 4.138864443899999
cv: 1.109697720577637
```

Exercise 2

A

The probability of observing at least one flood (p) in a given number of time units (n) is 1 - (the probability of seeing no floods (P))^n. This means we want to calculate $p = 1 - P^n$.

In the case of a 100 year flood $P = 0.99$, which gives us $p = 1 - 0.99^{10} = 0.0956$ for observing a 100 year flood in 10 years.

B

When performin simple linear regression on a data set we assume that the data conforms to the following properties:

- (1) Linearity:
  - If the data has a nonlinear trend it is rather dificult to get a good fit with linear regression.
  - We can generally tell if the data is somewhat linear by looking at a scatter plot of the data.
  - We can fix this by transforming the data before regression (e.g. x^2 or log(x)).
- (2) Normality:
  - Is the data normally distributed around a mean?
  - We can test this with a histogram, a qq-plot or with a KS-test.
  - This can also be fixed by transforming the data.
- (3) Independence:
  - Does a given data point depend on the previous data points? An example of dependent data would be sea temperature. The ocean has huge thermal mass, so the temperature is always dependent on yesterdays temperature.
  - We can test this by checking the datas auto correlation.

    - Another way this assumption can be broken is if there are several independent variables, and some of them depend on each other. The solution to this is generally to drop one of the variables.
- (4) Homoscedasticity:
    - Is the variance constant?
    - We can test this by looking at a scatter plot of the resuduals.
    - There is no good way to fix this, but the result from the linear regression should still be valid. We just cant depend on the standard error.

Of these assumptions it is clear that 2 and 4 are broken. It is hard to tell if 3 is broken, byt seeing as we are looking at flood data it seems unlikely. Floods are fairly independent of one another as far as i know. The data also seems fairly linear from 1A.

As for how to improve the analysis we have a problem. To fix the problems from 2 without breaking 1 is rather difficult. One thing that could be attempted is to perform multiple linear regressions on subsets of the data, with one regression from a runoff of 0-30, and another one from 30-250. Taking the logarithm of the data should also help with 2, but thenmaybe at the cost of 1.

A genereal solution would be to apply a different kind of regression, and circumvent the problem all together.

Exercise 3

A)

i)

We perform a t-test, with calculations in the "Gereal calculations" below.

The limits for a 95% confidence interval are: Lower limit: 143.3, upper limit: 146.7

The t-test is used because the true variance is unknown and n is rather small.

ii)

We perform a z-test, with calculations in the "Gereal calculations" below.

The limits for a 95% confidence interval are: Lower limit: 143.4, upper limit: 146.6

The z-test us used because the true variance is known.

B)

The confidence interval depends strongly on the variance. With the uncertainty that comes with an unknown true variance there is an increased uncertainty in the mean as well. The confidence interval therefore expands. This difference is small for large values of n, as a large sample size gives a good estimate of the true variance, which brings the result from the t-test closer to the z-test.

C)

We perform a chi squared test to find the confidence interval for the variance. Calculations can be seen in "General calculations" below. We can be 95% sure that the variance is between the limits: Lower limit: 16.08, upper limit: 25.56. As we can see there is quite a bit of uncertainty in the estimated variance, which explains the difference in A (i) and (ii).

In [2]:
```
#General calculations
import scipy as sc
import scipy.stats as scs

def getCI_unknown_V(alpha, mean, var, n):
    s_xb = sc.sqrt(var)/sc.sqrt(n)
    t = scs.t.ppf(1-alpha/2, n-1)
    l = mean - s_xb*t
    u = mean + s_xb*t

    return l, u

def getCI_known_V(alpha, mean, var, n):
    sig_xb = sc.sqrt(var)/sc.sqrt(n)
    z = scs.norm.ppf(1-alpha/2)
    l = mean - sig_xb*z
    u = mean + sig_xb*z

    return l, u

def getCI_V(alpha, var, n):
    chisq_1 = scs.chi2.ppf(1-alpha/2, n-1)
    chisq_2 = scs.chi2.ppf(alpha/2, n-1)
    lower = ((n-1)*var)/chisq_1
    upper = ((n-1)*var)/chisq_2

    return lower, upper

l, u = getCI_unknown_V(0.05, 145, 20, 30) # A i
l = float(l); u = float(u)
print(f"Lower limit: {l:.4}, upper limit: {u:.4}")

l, u = getCI_known_V(0.05, 145, 20, 30) # A ii
l = float(l); u = float(u)
print(f"Lower limit: {l:.4}, upper limit: {u:.4}")

l, u = getCI_V(0.05, 20, 145) # C
l = float(l); u = float(u)
print(f"Lower limit: {l:.4}, upper limit: {u:.4}")
```

```
Lower limit: 143.3, upper limit: 146.7
Lower limit: 143.4, upper limit: 146.6
Lower limit: 16.08, upper limit: 25.56
```

Exercise 4

A)

When training a machine learning model it optimises parameters to minimise the error in the estimate. This error can be calculated in many ways, but in some whay or another it represents the difference between the observed values, and the estimated values. This is known as the training error. If we do not separate the input data into a traning set and a test set this training error would be the absolute best case scenario for our model, and it would not be representative of real use. There is no practical reason to estimate data we allready have, so we want to see how well the model can predict values it has not seen before. This is where the test set comes in. By only training the model on a subset of the data and then testing it on a different set, we obtain a test error. This is a more representative value for the quality of the model. A common split percentage is around 67% to 33% train to test.

To be concise train test splits help reduce bias in the model.

B)

If the model becomes too complex we run the risk of overfitting. Most data can be approximated using a combination of random variables, but this is not a good approximation, as a train-test split would show. When we fit a machine learning algorithm we are interested in the trend of the data, not the noice, as the noice is random. With too many parameters we will start fitting the noice, which gives a worse result once the model is used on a different data set from the same source.

Exercise 5

A)

A simple linear regression with a t-test on the slope coefficient should be a suitable test. In this case there is heavy periodicity to the data, so we might want to make sure that we start and stop the segment we run the linear regression on at the same point in a phase, seeing as there will be a trend from two extremes in a preiod even though the data could be trend-less. This becomes less important as we have more data and more periods.

Another alternative is the Mann-Kendall trend test, which for all values compares the sign of the delta between the current value and all preceding values, and looks for a trend. This is done by computing:

$$S = \sum_{k-1}^{t-1} \sum_{j-k+1}^{t} sgn(x_j - x_k)$$

Where t is the index of the final point in the comparison and j and k are iteration variables. The value of S determines the trend with a positive S being a positive trend.

B)

At a first glance eiteher fft_A or fft_C could be the correct fourier transform, but there are two reasons why A is correct:

- 1: X_t is not completely uniform, so there should be several peaks, with one major one for the main signal. This fits with A.
- 2: The frequency of the signal in X_t is closer to 0.2 than 0.35, Whis fits with A