Candidate number: 21
Date: November 23, 2020
Exam Submission for Geophysical Data Science

In [2]:
```
#import modules before starting
import numpy as np
import pandas as pd
import scipy.stats as st
import seaborn as sn
import scipy
import datetime
import math
import sklearn
import statsmodels.formula.api as smf
import matplotlib.pyplot as plt
%matplotlib inline

import warnings
warnings.filterwarnings('ignore')

from scipy import stats
from scipy.stats import t
from scipy.stats import norm
from scipy.stats import f
from scipy.stats import chi2

from scipy.stats import hypergeom
from scipy.stats import binom
from scipy.stats import poisson
from scipy.stats import nbinom
from scipy.stats import geom

from scipy.special import comb
from statsmodels.formula.api import ols
```

Question 1: Random variable parameter estimation

(a) Find the expected value

In [6]:
```
Expected_Value = (-1 * 1/3) + (3 * 1/2) + (4 * 1/6)
Expected_Value
```

Out[6]: 1.8333333333333335

The expected value is around 1.83

(b) Find the variance

In [40]:
```
#The variance is E(x^2) - E(x)^2
#E_x2 = (-1^2) * 1/3 + 3^2 * 1/2 + 4^2 *1/6
E_x2 = ((-1)**2)/3 + ((3)**2)/2 + ((4)**2)/6
Ex_2 = Expected_Value**2
Variance = E_x2 - Ex_2
Variance, Ex_2, E_x2
```

Out[40]:
```
(4.138888888888888, 3.3611111111111116, 7.5)
```

The variance is around 4.14

(c) Find the mode

The mode is just the value that occurs most often, so it will be the value with the highest probability.
The mode is 3

(d) Find the coefficient of variation

The coefficient of variation (CoV) is the ratio of the standard deviation to the expected value

In [41]:
```
CoV = np.sqrt(Variance)/Expected_Value
CoV
```

Out[41]:
```
1.1096868741576094
```

The coefficient of variation is around 1.11

Question 2: Frequency analysis and linear regression

(a) What is the probability to observe at least one 100-years flood or larger within a period of 10 years?

Observed at least one = Opposite of observed none

In [13]:
```
P_100yrFlood = 1/100
print("The probability of a 100 yr flood occuring in a given year i
s:", P_100yrFlood)
```

```
The probability of a 100 yr flood occuring in a given year is: 0.0
1
```

In [14]:
```
P_observedNone_perYr = 1-P_100yrFlood
P_observedNone_10yrs = P_observedNone_perYr**10
print("The probability of a 100 yr flood not occuring in a 10 yr pe
riod is:", np.round(P_observedNone_10yrs,4))
```

```
The probability of a 100 yr flood not occuring in a 10 yr period i
s: 0.9044
```

In [17]:
```
print("The probability to observe at least one 100 yr flood in the
10 yrs period is:", np.round(1-P_observedNone_10yrs,4))
```

```
The probability to observe at least one 100 yr flood in the 10 yrs
period is: 0.0956
```

There is a ~9.6% chance to observe at least one 100-years flood within a period of 10 yrs

(b) Describe which assumption of a simple linear regression is violated in this analysis, and discuss strategies that can be used to improve the analysis

The assumptions of a simple linear regression are: Linearity, Normality, Homoscedasity, and Independence
Linearity means we assume there is a linear relationship between the dependent and independent variable. From the scatterplot in Figure 1a we can see that the relationship is somewhat linear, but clearly not fully linear. For example, the four points where run-off is around 160 yield vastly different flood levels from around 250 to 2000. However, it does not yield a flood level that is close to the median, so the high run-off is mapped to higher flood levels. Overall, I would say that the assumption of linearity is partly violated.
The QQ plot in figure 1B gives some information about the normality of the distribution. For a perfectly normal distribution the QQ plot would look like a straight diagonal line with a slope of 1. In figure 1B the line is pretty flattened out in the middle, so it looks like the normality assumption is violated.
Independence means that there isn't autocorrelation, which we can't really tell anything about from these graphs as there is no information about sequence of observations presented
It is a little complex to say if the data is really homoscedastic (i.e. constant variance accross the range of the independent variable), since we have little information about the residuals in these plots. but we can guess that it is probably not homoscedastic. There seems to be much more vertical spread as we move along the x axis, and indeed the regression residuals are larger when run-off is higher. So I would say homoscedasity is also, at least partly, violated.
In summary, linearity, normality, and homoscedasity appear at least partly violated from the graphs

Some strategies that could be used to improve the analysis could be to subset the data, and analyse it in parts. For example, analysing the higher values of runoff separately from the middle area might allow us to examine a region (in the middle) that is more linear, normmal, and homoscedastic than the entire dataset. We could do an influence analysis to determine if a few datapoints are having a lot of influence on the regression, and try to do the analysis without them included. We could try some transformations on the data, maybe see how doing a log-transformation influences the scatterplot and QQ plot, or try some other transformations to see if we can find one that makes the assumptions closer. We can try to think if there might be other important factors influencing the flood levels, and maybe try to gather more data and make a multiple linear regression. Also we could try to use a different method other than linear regression which does not have the same underlying assumptions.

Question 2: Confidence Intervals
A sample of 30 random observations produced a mean of 145 and variance of 20.
(a) What is the 95% confidence interval on the mean assuming a normal distribution if
(i) the true variance is unknown and estimated as 20
(ii) the true variance is 20

(a-i) For unknown variance I will use the student t distribution and the equations:
$$L = \bar{x} - t_{1-\frac{\alpha}{2},n-1}\, s_{\bar{x}}$$
$$U = \bar{x} + t_{1-\frac{\alpha}{2},n-1}\, s_{\bar{x}}$$

In [23]:
```
xbar = 145
variance = 20
n = 30
alpha = 0.05

stdev = np.sqrt(20)
sxbar = stdev / np.sqrt(n)

from scipy.stats import t

myt = t.ppf(1 - alpha/2, df=29)

L = xbar - myt*sxbar
U = xbar + myt*sxbar
print("Lower CI:", L)
print("Upper CI:", U)
```

```
Lower CI: 143.33007698998662
Upper CI: 146.66992301001338
```

The 95% confidence interval for unknown variance is 143.3 to 146.7

(a-ii) For known variance I will use the normal distribution and the equations:

$$L = \bar{x} - z_{1-\frac{\alpha}{2}}\sigma_{\bar{x}}$$

$$U = \bar{x} + z_{1-\frac{\alpha}{2}}\sigma_{\bar{x}}$$

In [26]:
```
from scipy.stats import norm
myz = st.norm.ppf(1 - alpha/2)

L2 = xbar - myz*sxbar
U2 = xbar + myz*sxbar
print("Lower CI:", L2)
print("Upper CI:", U2)
```

```
Lower CI: 143.39969610788157
Upper CI: 146.60030389211843
```

The 95% confidence interval for known variance is 143.4 to 146.6

(b) What is the reason for the difference of results in part (i) and part (ii)?

The difference is because of the distribution used. For unknown variance I used the t-distribution, which is a bit wider spread than the normal distribution. This is helpful because if we have a small sample size and/or our variance is unknown, we are less certain about our conclusions so the wider t-distribution better models our uncertainty. As n gets large the t-distribution gets closer to the normal distribution. This is why when we assume variance is unknown and use the t distribution, the confidence interval is a bit wider than if we assume the variance is known and use the normal distribution.

(c) What is the 95% confidence interval on the variance?

For this I will use the Chi-squared distribution and the following formulas:

$$L = \frac{(n-1)s_x^2}{\chi^2_{1-\alpha/2,n-1}}$$

$$U = \frac{(n-1)s_x^2}{\chi^2_{\alpha/2,n-1}}$$

In [29]:
```
from scipy.stats import chi2
chiL = chi2.ppf(1-alpha/2, n-1)
chiU = chi2.ppf(alpha/2, n-1)
```

In [31]:
```
Lvar = ((n-1)*variance)/chiL
Uvar = ((n-1)*variance)/chiU

print("Lower CI of variance:", Lvar)
print("Upper CI of variance:", Uvar)
print("Variance:", variance)
```

```
Lower CI of variance: 12.685280051047778
Upper CI of variance: 36.14366602272549
Variance: 20
```

The 95% confidence interval on the variance is 12.7 to 36.1

Question 4: Machine learning
(a) Why is it common to split the dataset into a training set and a test set when doing machine learning?
(b) In many machine learning algorithms you have a parameter that controls the complexity of the model. Why do we want to control this complexity?

(a) It is common to split the dataset into training and test sets because we need data to develop the model on (training set) and then we want to test the model on data that hasn't been used in model development to see if the model can predict data it wasn't specifically trained on (test set). Training error is the error of the model in predicting the training set. Test error is the error of the model in predicting the test set. Usually the training error is lower than the test error. However, in an overfitted model, the training error gets lower at the expense of the test error being higher than it would have been with a well-fitted model. This is because the model is too specific to the training set used, and does not really represent the whole data set. Having set aside a test set to test the model with allows us to compare models to see which ones best represents the whole data set and not just the training set used in model development. This makes us confident that our model will be more reliable and robust when presented with data that is new altogether (e.g. additional data collected after model development).

(b) We want to control the complexity to prevent overfitting and to have a model that is as parsimonious as possible. An overly complex model may be overfitted to the training data and be less robust when presented with new data. In the example of a multiple linear regression algorithm, adding in more parameters, even if they have low correlation with the dependent variable, will often result in a lower R^2 value, but at the expense of the model having true explanatory power. The coefficients of the parameters will be modified by having added in more parameters that are not really related to the dependent variable. Also, there is increased risk of correlation between the variables and thus violation of underlying assumptions. Other algorithms used for machine learning could have similar problems introduced by excess model complexity. Looking for a parsimonious model is essentially a statistical way of applying Occam's razor, with the goal of reducing unncessary model complexity to find one that best represents the data.

Question 5: Time series analysis and Fourier transformation
a) How could you test if there is a significant trend in Xt? Explain a suitable test.
One way to test for a trend would be to use a linear regression and a t-test. This method is relatively simple to interpret and represent, but has the downside compared to other tests (such as test Mann-Kendall test) of having the underlying asssumption of normally distributed residuals. To do this test I would first fit a linear regression with time (T) as the independent variable in the form: Y = a + bT
I would then use a t-test to determine if the coefficient b is significantly different from zero. If b is significantly different from 0 there is a trend.
In my hypothesis test my null hypothesis is that there is no trend, and the alternative hypothesis is that there is a trend.

b) which graph shows the Fourier transform?

I would say that graph A shows the Fourier transform of Xt. I think this is the case because there is a clear periodicity in the graph of Xt. It occurs at a period of around 5 seconds. The Fourier transformation would show this as a magnitude of the frequency (1/period = 1/5 = 0.2), which we see in the tall peak in graph A. On the opposite side of the Nyquist frequency we can see a mirroring of the 0.2 frequency occuring at 0.8.

In [ ]: