

Assignment 3

$$h(x) = f(2x) \cdot (g(x))^2$$

I set: $f(2x) = u(x)$

$$[g(x)]^2 = v(x)$$

$$\Rightarrow h(x) = u(x) \cdot v(x)$$

$$\Rightarrow h'(x) = u'(x) \cdot v(x) + u(x) \cdot v'(x)$$

We have: $u'(x) = [f(2x)]' = 2 \cdot f'(2x)$

$$v'(x) = [g(x)^2]' = 2 \cdot g(x) \cdot g'(x)$$

$$\Rightarrow h'(x) = 2 \cdot f'(2x) \cdot (g(x))^2 + f(2x) \cdot 2 \cdot g(x) \cdot g'(x)$$

$$\Rightarrow h'(1) = 2 \cdot f'(2) \cdot (g(1))^2 + f(2) \cdot 2 \cdot g(1) \cdot g'(1)$$

$$= 2 \cdot 2 \cdot (35)^2 + 2 \cdot f(2) \cdot 35 \cdot (-2)$$

$$= \frac{4900}{6+25} - 140 f(2)$$

We have: $f(2) = f(1) + \int_1^2 f'(x) dx$

$$= 3 + \frac{(4+2) \cdot 1}{2} = 3 + 3 = 6$$

$$\Rightarrow h'(1) = \frac{4900}{6+25} - 140 \cdot 6 = \frac{4900}{31} - 840$$

2)

a) $P_r(4 < x < 6)$

$$= \int_4^6 f(x) dx = \int_4^5 f(x) dx + \int_5^6 f(x) dx$$

$$= \frac{2}{10} \Big|_4^5 + \int_5^6 f(x) dx = \frac{2}{10} + \left(\frac{2}{10} + \frac{1}{10} \right) \cdot 2 = 0,2 + 0,3 = 0,5$$

$$b) E[X] = \int_{-\infty}^{+\infty} x f(x) dx$$

We will break the integral into 3 parts:

$$E[X] = \int_0^2 x f(x) dx + \int_2^5 x f(x) dx + \int_5^7 x f(x) dx$$

$$= \int_0^2 x \cdot \frac{1}{10} dx + \int_2^5 x \cdot \frac{2}{10} dx + \int_5^7 \left(\frac{7}{10} - \frac{x}{10} \right) x dx$$

$$= \frac{1}{10} \cdot \frac{2^2}{2} + \frac{2}{10} \left(\frac{5^2 - 2^2}{2} \right) + \frac{7}{10} \left(\frac{7^2 - 5^2}{2} \right) - \frac{1}{10} \left(\frac{7^3}{3} - \frac{5^3}{3} \right)$$

$$= \frac{103}{30} \approx 3,43$$

c) We need to calculate

$$E[X^2] = \int_0^2 x^2 \cdot \frac{1}{10} dx + \int_2^5 x^2 \cdot \frac{2}{10} dx + \int_5^7 x^2 \left(\frac{7}{10} - \frac{x}{10} \right) dx$$

$$= \frac{1}{10} \left(\frac{2^3 - 0^3}{3} \right) + \frac{2}{10} \left(\frac{5^3 - 2^3}{3} \right) + \frac{7}{10} \left(\frac{7^3 - 5^3}{3} \right) - \frac{1}{10} \left(\frac{7^4 - 5^4}{4} \right)$$

$$= \frac{218}{15}$$

$$\Rightarrow \text{Var}(X) = E[X^2] - (E[X])^2$$

$$= \frac{218}{15} - \left(\frac{103}{30} \right)^2 = \frac{2491}{900} \approx 2,745$$

37

a) From Mac & Laurin series for e^x and $\log(1+x)$ with degree 5

$$\log(1+2x) = 2x - 2x^2 + \frac{8x^3}{3} - \frac{16x^4}{4} + \frac{32x^5}{5}$$

$$\text{and } e^{-x^2} = 1 - x^2 + \frac{x^4}{2}$$

The degree 5 Maclaurin polynomial

$$k(x) = x \log(1+2x) e^{-x^2}$$

$$= x \cdot \left(2x - 2x^2 + \frac{8x^3}{3} - \frac{16x^4}{4} + \frac{32x^5}{5} \right) \cdot \left(1 - x^2 + \frac{x^4}{2} \right)$$

$$= 2x^2 - 2x^3 + \frac{2x^4}{3} - 2x^5$$

$$b) k^{(5)}(0) = (-2) \cdot 5! = -240$$

$$c) \lim_{x \rightarrow 0} \frac{k(x) - 2x^2}{x^3} = \lim_{x \rightarrow 0} \frac{k(x)}{x^3} = \frac{2x^2 - 2x^3 + \frac{2x^4}{3} - 2x^5 - 2x^2}{x^3}$$

$$= \lim_{x \rightarrow 0} \frac{x^3 \cdot \left(-2 + \frac{2x^3}{3} - 2x^3 \right)}{x^3} = \lim_{x \rightarrow 0} \left(-2 + \frac{2x^3}{3} - 2x^3 \right) = -2$$

$$\Rightarrow \lim_{x \rightarrow 0} \frac{k(x) - 2x^2}{x^3} = -2$$

Assign 4 Question 3.

You should submit this entire notebook for marking. If you have results but are missing code that generated those results you will lose marks. As well as Computer Prac 3, you may need to refer to the course material on continuous distributions https://myuni.adelaide.edu.au/courses/95091/pages/week-5-common-continuous-random-variables-in-data-science?module_item_id=3464136, and the reference materials for some particular distributions in Python: <https://docs.scipy.org/doc/scipy/reference/stats.html>

You have taken up record collecting, and your favourite online record seller, *StatsVinyl*, are vinyl record fans who are also very statistically minded, so they give detailed statistics on their sales.

(a) Suppose that the number 1 record on your want list has a pricing guide given by a normal distribution with a mean of \$200 and standard deviation of \$50. Define a normal random variable which represents the price of this record and use it to calculate the probability that the record would be sold for less than \$120

```
In [4]: import scipy.stats as stats
```

```
In [5]: # Code for part (a) here.
# at the end you should use a print statement to present your answer in a sentence

# Define variables
mean_price = 200
std_price = 50
# Create normal distribution for the price
price_distribution = stats.norm(loc=mean_price, scale=std_price)
# Probability that the price is less than 120
p = price_distribution.cdf(120)
print(f"Probability the price < 120 dollar: {p}")
```

Probability the price < 120 dollar: 0.054799291699557974

(b) They also model the availability of records. The waiting time for a copy of the record you want is modelled by an exponential random variable, where it is expected that on average you would have to wait 3 months for a copy to come in. Define an exponential random variable representing the waiting time for a copy of this record and use it to determine the probability that you a copy will come in after only 1 month or less.

```
In [8]: # Code for part (b) here.
# at the end you should use a print statement to present your answer in a sentence

# Define variables
mean_wait = 3 # months
# Create exponential distribution for waiting time
wait_distribution = stats.expon(scale=mean_wait)
# Probability that the waiting time is less than 1
p = wait_distribution.cdf(1)
print(f"Probability wait < 1 month: {p}")
```

Probability wait < 1 month: 0.28346868942621073

(c) Now you are wondering, how long can you expect the wait to be with a 90% probability? That is, if your random variable is X , what is x such that $Pr(X < x) = 0.9$.

In the past we have used a while loop to solve a problem of this sort, but instead here we will use some built-in functions. What we need is an inverse of the cdf, however instead the probability distributions have an inverse of the survival function (sf) which is defined by $sf(x) = 1 - cdf(x)$. The inverse survival function (isf) is the inverse function of the survival function. This is a method associated with a continuous random variable, just as cdf is. Use the isf to solve the problem above.

```
In [10]: # Code for part (c) here.
# at the end you should use a print statement to present your answer in a sentence
# Calculate the waiting time for a 90% probability

months = wait_distribution.isf(0.1)
print(f"Waiting months for a 90% probability: {months}")
```

Waiting months for a 90% probability: 6.907755278982137