ExamAugust_2023_problem

November 15, 2023

1 Exam 15th of August 2023, 8.00-13.00 for the course 1MS041 (Introduction to Data Science / Introduktion till dataanalys)

1.1 Instructions:

1. Complete the problems by following instructions.
2. When done, submit this file with your solutions saved, following the instruction sheet.

This exam has 3 problems for a total of 40 points, to pass you need 20 points.

1.2 Some general hints and information:

- Try to answer all questions even if you are uncertain.
- Comment your code, so that if you get the wrong answer I can understand how you thought this can give you some points even though the code does not run.
- Follow the instruction sheet rigorously.
- This exam is partially autograded, but your code and your free text answers are manually graded anonymously.
- If there are any questions, please ask the exam guards, they will escalate it to me if necessary.

1.3 Tips for free text answers

- Be VERY clear with your reasoning, there should be zero ambiguity in what you are referring to.
- If you want to include math, you can write LaTeX in the Markdown cells, for instance `$f(x)=x^2$` will be rendered as $f(x) = x^2$ and `$$f(x) = x^2$$` will become an equation line, as follows

  $$f(x) = x^2$$

  Another example is `$$f_{Y \mid X}(y,x) = P(Y = y \mid X = x) = \exp(\alpha \cdot x + \beta)$$` which renders as

  $$f_{Y \mid X}(y,x) = P(Y = y \mid X = x) = \exp(\alpha \cdot x + \beta)$$

1.4 Finally some rules:

- You may not communicate with others during the exam, for example:
  - You cannot ask for help in Stack-Overflow or other such help forums during the Exam.
  - You may not communicate with AI's, for instance ChatGPT.
  - Your on-line and off-line activity is being monitored according to the examination rules.

1.5 Good luck!

[ ]:
```python
# Insert your anonymous exam ID as a string in the variable below
examID="XXX"
```

1.6 Exam vB, PROBLEM 1

Maximum Points = 14

A courier company operates a fleet of delivery trucks that make deliveries to different parts of the city. The trucks are equipped with GPS tracking devices that record the location of each truck at regular intervals. The locations are divided into three regions: downtown, the suburbs, and the countryside, however there is always the possibility the truck breaks down and it goes to the workshop. The following table shows the probabilities of a truck transitioning between these regions at each time step:

Current region	Probability of transitioning to downtown	Probability of transitioning to the suburbs	Probability of transitioning to the countryside	Probability of transitioning to the Workshop
Downtown	0.3	0.7	0	0
Suburbs	0.2	0.5	0.3	0
Countryside	0	0	0.5	0.5
Workshop	0	0	0	1

1. If a truck is currently in the downtown, what is the probability that it will be in the countryside region after 10 time steps? [2p]
2. If a truck is currently in the downtown, what is the probability that it will be in the countryside region **the first time** after three time steps or more? [2p]
3. Is this Markov chain irreducible? Explain your answer. [3p]
4. What is the stationary distribution? Furthermore it it reversible? (Explain your answer) [3p]
5. Advanced question: What is the expected number of steps it takes starting from the Downtown region to first reach the Workshop? Hint: to get within 1 decimal point, it is enough to compute the probabilities for hitting times below 50. Motivate your answer in detail [4p]. You could also solve this question by simulation, but this gives you a maximum of [2p].

[ ]:
```python
# Part 1

# Fill in the answer to part 1 below
problem1_p1 = XXX
```

[ ]:
```python
# Part 2

# Fill in the answer to part 2 below
problem1_p2 = XXX
```

1.7 Part 3

Double click this cell to enter edit mode and write your answer for part 3 below this line.

[ ]:

```
# Part 3

# Fill in the answer to part 3 below as a boolean
problem1_irreducible = True/False
```

[ ]:

```
# Part 4

# Fill in the answer to part 4 below
# the answer should be a numpy array of length 3
# make sure that the entries sums to 1!
problem1_stationary = XXX
problem1_reversible = True/False
```

2 Part 4

Double click this cell and write your motivation below this line

2.1 Part 5

Double click this cell to enter edit mode and write your answer for part 5 below this line.

[ ]:

```
# Part 5

# Fill in the answer to part 5 below
# That is, the expected number of steps
problem1_ET = XXX
```

2.2 Exam vB, PROBLEM 2

Maximum Points = 13

You are given a “Data Science Salaries” dataset found in `data/salaries.csv`, which contains employment information of data scientists up to 2023 and the salary obtained. Your task is to train a `linear regression` model to predict the salary of a data scientist based on the employment information.

To evaluate your model, you will split the dataset into a training set and a testing set. You will use the training set to train your model, and the testing set to evaluate its performance.

`Experience level`: 0 = Entry Level, 1 = Mid Level, 2 = Senior Level, 3 = Executive Level.

`Employment type`: 0 = Part Time, 1 = Full Time, 2 = Contractor, 3 = Freelancer

1. Load the data into a pandas dataframe `problem2_df`. Based on the column names, figure out what are the features and the target and fill in the answer in the correct cell below. [1p]
2. Split the data into train and test. [1p]

3. Train the model. [1p]
4. Come up with a reasonable metric and compute it. Provide plots that show the performance of the model. Reason about the performance. [4p]
5. Predict the 2023 salary of a data scientist that works full time (1) at mid employment level (1) with 0 remote ratio. Then, looking at the output of `problem2_model.coef_`, which are the coefficients of the linear model, would a higher remote ratio result in a higher predicted salary or vice versa? [3p]
6. Advanced question: On the test set, plot the empirical distribution function of the residual with confidence bands (i.e. using the DKW inequality and 95% confidence). What does the confidence band tell us? What can the confidence band be used for? [3p]

[ ]:

```
# Part 1
# Let problem2_df be the pandas dataframe that contains the data from the file
# data/abalone.csv
problem2_df = XXX
```

[ ]:

```
# Part 1

# Fill in the features as a list of strings of the names of the columns

problem2_features = ["XXX"]

# Fill in the target as a string with the correct column name

problem2_target = "XXX"
```

[ ]:

```
# Part 2


# Split the data into train and test using train_test_split
# keep the train size as 0.8 and use random_state=42
problem2_X_train,problem2_X_test,problem2_y_train,problem2_y_test = XXX
```

[ ]:

```
# Part 3

# Include the necessary imports

# Initialize your linear regression model
problem2_model = XXX

# Train your model on the training data
```

2.3 Part 4

Double click this cell to enter edit mode and write your answer for part 4 below this line.

[ ]:

```
# Part 4
# Write the code to diagnose your model
```

2.4 Part 5

Double click this cell to enter edit mode and write your answer for part 5 below this line.

[ ]:
```
# Part 5

# Put the code for part 5 below this line
```

[ ]:
```
# Part 5

problem2_predicted_salary = XXX
```

2.5 Part 6

Double click this cell to enter edit mode and write your answer for part 6 below this line.

[ ]:
```
# Part 6

# Put the code for part 6 below this line
```

2.6 Exam vB, PROBLEM 3

Maximum Points = 13

2.7 Random variable generation

1. [4p] Using inversion sampling, construct 1000 samples from the below distribution

$$F[x] = \begin{cases} 0, & x \leq 0 \\ e^x - 1, & 0 < x < \ln(2) \\ 1, & x \geq \ln(2) \end{cases}$$

2. [2p] Use the above 1000 samples to estimate the mean and variance
3. [4p] Using the **Accept-Reject** sampler (**Algorithm 1** in TFDS notes) construct 1000 samples from the same distribution, what proposal distribution did you choose and why? What proportion of samples where accepted?
4. [3p] Explain if it is possible to sample from the density

$$f(x) = Ce^{-(x^2-2)^2}$$

   using the **Accept-Reject** sampler (**Algorithm 1** in TFDS notes) with sampling density given the Gaussian. Here C is a constant to make sure that f is a density, it between roughly 1.34 and 1.35.

[52]:
```
# Part 1
# Write your code below
```

[ ]:

```
# Part 1

# Put the resulting samples into the following variable

problem3_samples = XXX
```

[ ]:

```
# Part 2

problem3_mean = XXX

problem3_variance = XXX
```

[ ]:

```
# Part 3

# Write your code to solve the problem below
```

[ ]:

```
# Part 3

# Write your answer in this cell below

problem3_samples_accept_reject = XXX

# Put your answer for the proportion of samples accepted below

problem3_acceptance_rate = XXX
```

3 Part 3

Double click this cell and write you answer for part 3 below, explain what proposal distibution you chose and why you can choose it.

4 Part 4

Double click this cell and write you answer for part 4 below, explain if the Gaussian works as a proposal distribution for this density.