ExamJune_2023_problem

June 13, 2023

1 Exam 14th of June 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.
- I (Benny) will visit the exam room at around 10:30 to see if there are any questions.

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!

[ ]:

```
# 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. 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
Downtown	0.3	0.7	0
Suburbs	0.2	0.5	0.3
Countryside	0	0.5	0.5

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? [3p]
5. Advanced question: What is the expected number of steps it takes starting from the Downtown region to first reach the Countryside region and then returning to Downtown. Hint: to get within 1 decimal point, it is enough to compute the probabilities for hitting times below 120. Motivate your answer in detail [4p]. You could also solve this question by simulation, but this gives you a maximum of [2p].

[ ]:

```
# Part 1

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

[ ]:

```
# 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
```

1.8 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
```

1.9 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
```

1.10 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
```

1.11 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
```

1.12 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
```

1.13 Exam vB, PROBLEM 3

Maximum Points = 13

For this problem we have the `Diabetes` dataset, I have encoded the categorical features using One-Hot encoding, namely the following `['smoking_No Info', 'smoking_current', 'smoking_ever', 'smoking_former', 'smoking_never', 'smoking_not current', 'sex_Female', 'sex_Male', 'sex_Other']`.

Treating this as a classification problem, we will train a logistic regression model to predict whether the patient has diabetes or not. Then the task is to evaluate the model and using it to make some conclusions.

Instructions:

1. Load the file `data/diabetes.csv` into the pandas dataframe `problem3_df`. Decide what should be features and target, give motivations for your choices. [3p]
2. Create the `problem3_X` and the `problem3_y` as numpy arrays with `problem3_X` being the features and `problem3_y` being the target. Do the standard train-test split with 80% training data and 20% testing data. Store these in the variables defined in the cells. [2p]
3. Now train a Logistic regression model on the training data. Using `sklearn.linear_model.LogisticRegression`. Hint: If you use many of the One-Hot encoded features you will probably see a warning about max iterations reached, adjust the hyperparameter `C` (this is the penalization) when you create your LogisticRegression.[2p]
4. Evaluation: Calculate the precision and recall for class 0 and 1 with 95% confidence bounds. Explain their meaning [3p]
5. Advanced question: Come up with a way to define the one-hot encoded feature that is most important for the prediction. Motivate your choice. [3p]

1.14 Part 1

Double click this cell to enter edit mode and write your answer for part 1 below this line.

What features are reasonable?

In regards to how much data we have, how many features do you think we should aim for?

What other features would you like to have used but was not collected?

Discussion

[ ]:
```
# Part 1

# Let problem3_df be the pandas dataframe that contains the data from the file
# data/visits_clean.csv
problem3_df = XXX
```

[ ]:
```
# Part 1

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

problem3_features = ["XXX"]

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

problem3_target = "XXX"
```

[ ]:
```
# Part 2

# Fill in your X and y below
problem3_X = XXX
problem3_y = XXX

# Split the data into train and test using train_test_split
# keep the train size as 0.8 and use random_state=42
problem3_X_train, problem3_X_test, problem3_y_train, problem3_y_test = XXX
```

[ ]:
```
# Part 3

# Initialize your LogisticRegression model
problem3_model = XXX

# Fit your initialized model on the training data
```

1.15 Part 4

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

[ ]:
```
# Part 4
# Give the answer for each of the following quantities in the form of a tuple
# Example, if we want to say that the precision for class 0 is between 0.31 and 0.69
# then we would answer
# problem3_precision_0 = (0.31,0.69)

problem3_precision_0 = XXX
problem3_recall_0 = XXX
problem3_precision_1 = XXX
problem3_recall_1 = XXX
```

1.16 Part 5

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

[ ]:

```
# Part 5

# Put whatever calculations you need here
```