260925886_Assignment_4

March 16, 2023

NL2DS - Winter 2023

Assignment 4 – Psycholinguistic data and regression

Name: **Usman Hameed Ullah**

Student ID: **260925886**

In this assignment we will work with several real psycholinguistics datasets. These datasets are inspired by discussion in this book by McGill Linguistics Professor Morgan Sondregger.

The first part of the problem set will examine some lexical decision data. You can read about lexical decision experiments in the wikipedia article here. (The first dataset also contains so-called *speeded naming* data. You can read about that in the speeded naming section of the first paper.)

The collection of the lexical decision data is originally described in.

Balota, D. A., Cortese, M. J., Sergent-Marshall, S. D., Spieler, D. H., and Yap, M. J. (2004). Visual word recognition of single-syllable words. Journal of Experimental Psychology: General, 133(2):283–316.

In the following paper, this data was reanalyzed using some new features (predictors).

R. H. Baayen, L. Feldman, and R. Schreuder. Morphological Influences on the Recognition of Monosyllabic Monomorphemic Words. Journal of Memory and Language, 53:496– 512, 2006. You can find a copy of this paper.

This data is discussed in Harald Baayen's book on linguistic data analysis.

Baayen, R. H. (2008). Analyzing Linguistic Data: A practical introduction to statistics. Cambridge University Press.

This particular file was derived from the original data available as as the `english` dataframe of the languageR package.

Copy the data to your drive folder from: https://drive.google.com/file/d/19ybVdUWwZh_hSw69DRG7rYJd8o5dK

[1]:
```python
from google.colab import drive
drive.mount('/content/drive/')
!ls "/content/drive/My Drive/english.csv"
```

```
Drive already mounted at /content/drive/; to attempt to forcibly remount, call
drive.mount("/content/drive/", force_remount=True).
'/content/drive/My Drive/english.csv'
```

Problem 1

Use Pandas to read the CSV file into a dataframe and then have a look at the data set.

[2]:
```python
import pandas as pd
from IPython.display import display

# Problem 1: Display the dataset.

english = pd.read_csv('/content/drive/My Drive/english.csv')
display(english)
```

```
      RTlexdec  RTnaming  Familiarity    Word AgeSubject WordCategory  \
0     6.543754  6.145044         2.37     doe      young            N
1     6.304942  6.143756         5.60  stress      young            N
2     6.424221  6.131878         3.87    pork      young            N
3     6.450597  6.198479         3.93    plug      young            N
4     6.531970  6.167726         3.27    prop      young            N
…          …         …            …       …        …              …
4561  6.753998  6.446513         2.40     jag        old            V
4562  6.711022  6.506979         3.17    hash        old            V
4563  6.592332  6.386879         3.87    dash        old            V
4564  6.565561  6.519884         4.97   flirt        old            V
4565  6.667300  6.496624         3.03    hawk        old            V

      WrittenFrequency  WrittenSpokenFrequencyRatio  FamilySize  \
0             3.912023                     1.021651    1.386294
1             6.505784                     2.089356    1.609438
2             5.017280                    -0.526334    1.945910
3             4.890349                    -1.044545    2.197225
4             4.770685                     0.924801    1.386294
…                  …                           …          …
4561          2.079442                    -1.686399    1.386294
4562          3.663562                     0.436718    1.609438
4563          5.043425                     0.504395    1.945910
4564          3.135494                     0.062801    1.945910
4565          4.276666                     1.049822    1.945910

      DerivationalEntropy  …    ConfbN  NounFrequency  VerbFrequency  CV  \
0                 0.14144  …  8.833900             49              0   C
1                 0.06197  …  5.817111            565            473   C
2                 0.43035  …  2.564949            150              0   C
3                 0.35920  …  0.000000            170            120   C
4                 0.06268  …  2.197225            125            280   C
…                    …   …        …              …             …   ..
4561              0.30954  …  0.000000             10              7   C
4562              0.15110  …  0.693147             38              7   C
4563              0.63316  …  0.693147            113            231   C
```

```
4564            0.99953  …  4.304065          10            66   C
4565            0.95422  …  5.552960         109            47   C

     Obstruent  Frication      Voice  FrequencyInitialDiphoneWord  \
0         obst      burst     voiced                    10.129308
1         obst  frication  voiceless                    12.422026
2         obst      burst  voiceless                    10.048151
3         obst      burst  voiceless                    11.796336
4         obst      burst  voiceless                    11.991567
…          …         …         …                           …
4561      obst  frication     voiced                     8.311644
4562      obst  frication  voiceless                    12.567203
4563      obst      burst     voiced                     8.920923
4564      obst  frication  voiceless                    10.425639
4565      obst  frication  voiceless                     9.054388

     FrequencyInitialDiphoneSyllable  CorrectLexdec
0                          10.409763             27
1                          13.127395             30
2                          11.003649             30
3                          12.163092             26
4                          12.436772             28
…                              …            …
4561                        8.390041             29
4562                       12.665546             29
4563                        9.287764             29
4564                       10.932142             29
4565                        9.148252             30

[4566 rows x 36 columns]
```

Question 1: Your first job is to familiarize yourself with the dataset by briefly examining the two papers above. First, read the wikipedia article on lexical decision, and briefly (2-4 sentences) explain the lexical decision experimental task.

Q1: put your answer here (please keep it brief, 2-3 sentences)

The lexical decision task is a commonly used experimental paradigm in cognitive psychology, which involves presenting participants with a string of letters and asking them to decide whether the string of letters is a real word or a non-word. The task measures the accuracy and speed at which people classify stimuli as words or non-words. It can be used to investigate how different factors such as word frequency, semantic priming, and context influence word recognition.

Start with the earlier paper then move on to the later paper. Note these two papers are long and use a lot of technical jargon from the field of psycholinguistics. *Reading each paper carefully would take several hours and you probably would not be able to understand everything unless you have previous familiarity with experimental psychology.* This is not the goal of this part of the assignment. Instead, the goal is to just familiarize yourself as efficiently as possible with what some of the columns in the data set mean. An important skill in data science is quickly evaluating the high level idea and questions studied in a paper and finding the places where quantitites are defined, without doing a careful reading.

A good way to approach this is to first read the abstract, the introduction and the conclusion and then have a look at the figures, always keeping in mind the data from the CSV above and trying to find interpretations for the various columns. Don't get stuck on stuff you don't understand unless you are pretty sure you need to understand it to answer the question.

Focus on figuring out where you can find the relevant information to answer the following questions.

Question 2: In these studies and in this dataset various regression models are used to analyze the experimental data. What was measured in these studies that corresponds to $\mathbf{y}$ in our notation from class (i.e., the quantities to be predicted) and which columns in the dataset have these values?

Q2: put your answer here (please keep it brief, 2-3 sentences)

The reaction time of participants in lexical judgement and speeded name tasks were measured in these studies and as such correspond to $\mathbf{y}$ in our notation from class. These correspond to RTlexdec and RTnaming in english.csv.

Question 3: In both papers a number of different quantities are used as predictors for the experimental measures. These correspond to the columns of our $\mathbf{X}$ matrix from class. Note that between these two papers there are a lot of variables, and this a lot of columns in the table. Please determine the meaning of the first seven features from these papers (`Familiarity` to `FamilySize`).

Q3: put your answer here (please keep it brief, 1-2 sentences/predictor)

Familiarity is the rating of how familiar English speakers are with the word.

Imageability describes how easily a mental image of the word can be formed.

WrittenFrequency is the frequency of the word in the English language.

OrthographicNeighbors is the number of words you can get by changing a single letter in a given word.

PhonologicalNeighbors is the number of words that sound similar to a given word

NumberofMorphemes describes how meaningful the word is by the number of meaningfull parts e.g. dog has one morpheme while dogs has two, dog+s

FamilySize is the number of words that can be traced back to a common ancestor.

Question 4: For each of these predictors, how would you intuitively expect it to relate to the reactions times in the $\mathbf{y}$ variables? (Note that there is no right or wrong answer here, so long as you give a justification for your reasoning). Please be brief, no more than 2-4 sentences per predictor.

Q4: put your answer here (please keep it brief, 1-2 sentences/predictor)

High **familiarity** should allow words to be more easily recognized by the brain therefore should lead to faster reaction times.

High **imageability** means its easier to form an image assosciated to the word so should lead to faster reaction times.

High **written frequency** would mean that these words are more commonly seen so should lead to higher reaction times.

Greater number of **orthographic neighbours** means that there are a lot more words very close to the given word so it can take longer for the brain to process and identify it so would lead to slower reaction times.

Similarly, greater number of **phonological neighbours** would also lead to slower reaction times since a lot of words sound the same so can take the brain longer to process.

Greater **number of morphemes** implies that word has many complex forms so the brain will be slower to process it and identify the correct one so would lead to slower reaction times.

Lastly, greater **family size** allows the brain to easily correlate various words to eachother that can be traced back to similar origin/meaning making reaction time faster.

Let's simplify the dataset a bit, to have fewer columns.

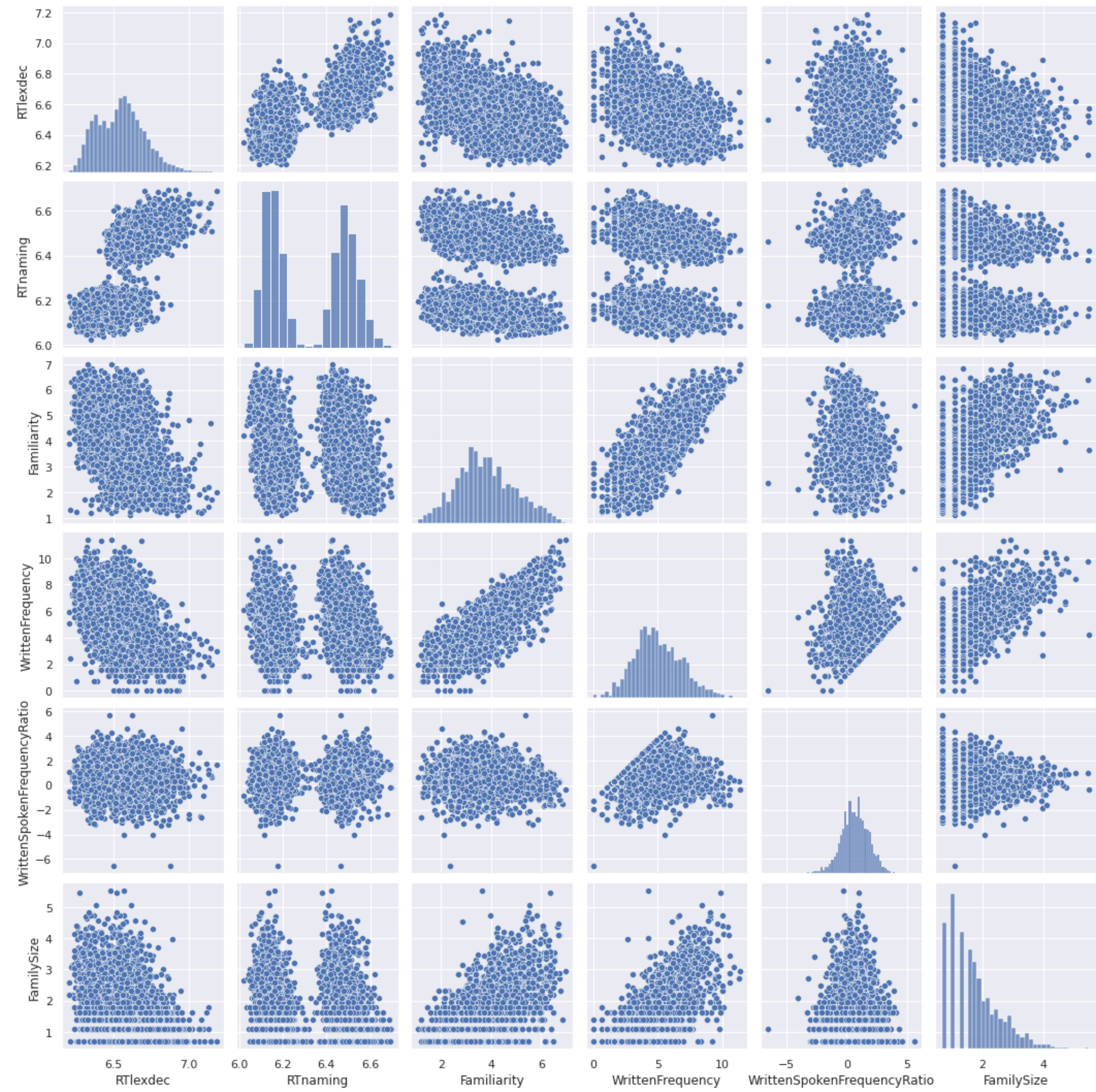
Problems 2-5

[5]:
```
# Problem 2: Write some code that drops all of the columns from the
# English dataset past the 9th column (the last column should be FamilySize)

english = english.iloc[:, :9]
```

Now use the Seaborn library to produce a set of plots between (see `pairplot`) all the variables in the dataset.

[6]:
```
import seaborn as sns; sns.set()
sns.pairplot(english)
```

[6]: `<seaborn.axisgrid.PairGrid at 0x7f375fca6bb0>`



Let's focus on the subset of the data from young participants.

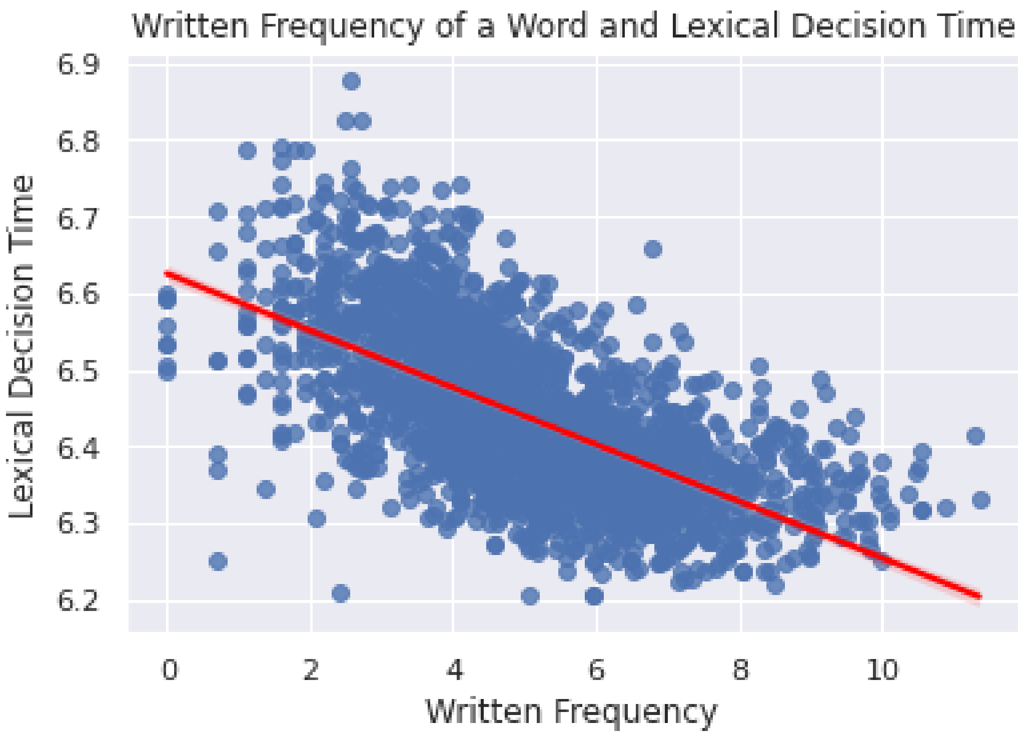
[7]:
```
# Problem 3: Write an expression that selects the subset of rows which correspond to the young speakers.

english_young = english.loc[english['AgeSubject'] == 'young']
```

Let's examine the relationship between the written frequency of a word on it's lexical decision time. Use `seaborn.regplot` to make a plot with a linear trend line that has the fequency on the x-axis and lexical decision time on the y-axis.

You may also find this page useful in understanding how to use Seaborne to plot regression lines: https://seaborn.pydata.org/tutorial/regression.html

[8]:
```
#Problem 4: use seaborne.regplot to plot the data with a best fit regressions␣
↪line.
import matplotlib.pyplot as plt

sns.regplot(x='WrittenFrequency', y='RTlexdec', data=english_young,␣
↪line_kws={'color': 'red'})
plt.xlabel('Written Frequency')
plt.ylabel('Lexical Decision Time')
plt.title('Written Frequency of a Word and Lexical Decision Time')
```

[8]:
```
Text(0.5, 1.0, 'Written Frequency of a Word and Lexical Decision Time')
```

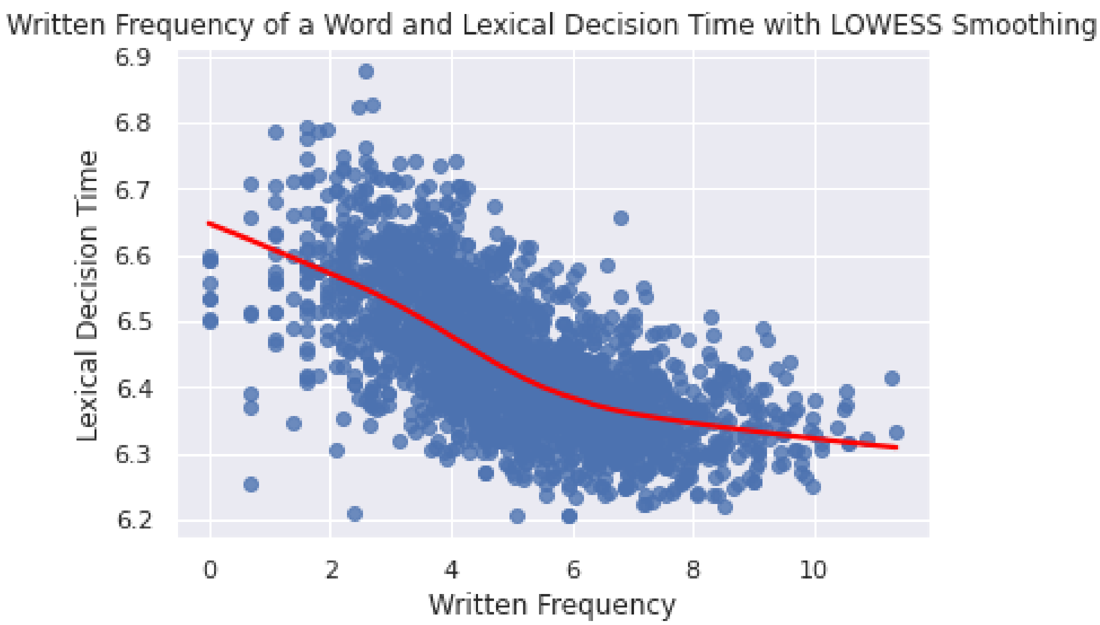


In order to examine wether or the relationship between two variables is really linear, it is useful to look at a *locally-smoothed regression line* that relates the x and y axes of a plot. This is a kind of regression model where the function is refit localy for many subsets of the data then a smooth line is interpolated between these points. One standard technique for this is known as *locally weighted scatterplot smoothing* or LOWESS and is implemented as an option for the line drawn by the `regplot` function. Using this examine whether the relationship between frequency and lexical decision times really looks linear.

[9]:
```
#Problem 5: use seaborne.regplot to mnake a plot with the best fit LOWESS line.
```

```
sns.regplot(x='WrittenFrequency', y='RTlexdec', data=english_young,␣
    ↪lowess=True, line_kws={'color': 'red'})
plt.xlabel('Written Frequency')
plt.ylabel('Lexical Decision Time')
plt.title('Written Frequency of a Word and Lexical Decision Time with LOWESS␣
    ↪Smoothing')
```

[9]: Text(0.5, 1.0, 'Written Frequency of a Word and Lexical Decision Time with LOWESS Smoothing')



Question 5: What do you see in this data when you look at the two plots above? Do you think that a linear model represents the relationship between written frequency and reaction times? It may be useful to turn on an off the plotting of the underlying data points with the `scatter=False` argument to the functions.

Q5: put your answer here

Examining the two plots above, it seems that with LOWESS smoothing applied, the curve is roughly a straight line and very close to the curve without smoothing applied. Therefore, the linear model does represent the relationship between written frequency and reaction times.

Let's try looking at some more complex models of the relationship between frequency and lexical decision time. Here is some starter code similar to those that we looked at in class for writing a polynomial regression. Complete the code as indicated.

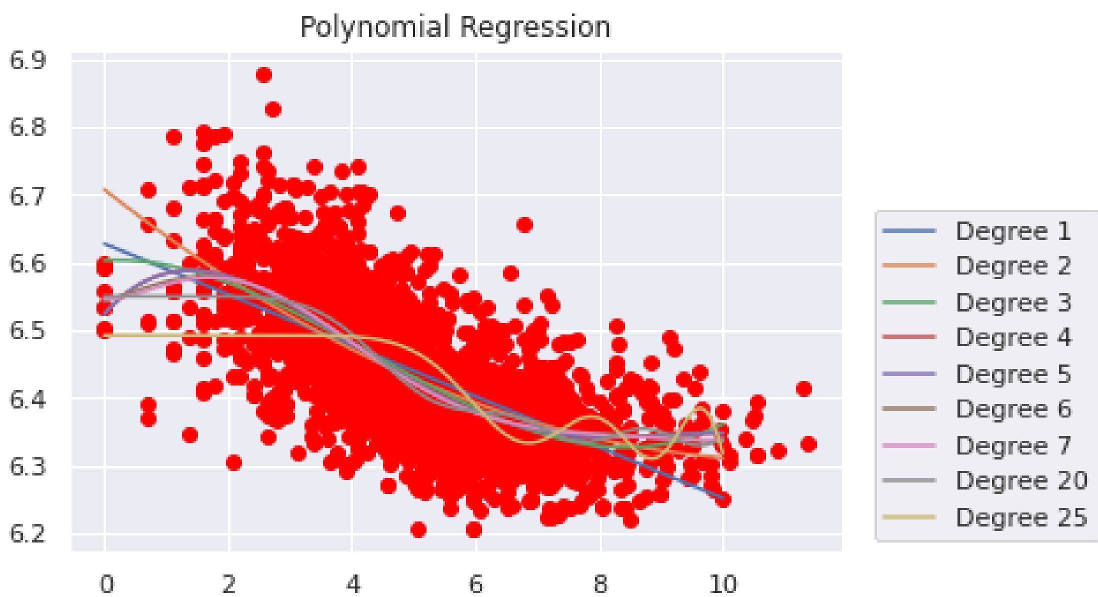
Problems 6-9

[10]:

```python
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
from sklearn.pipeline import make_pipeline
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import numpy as np

# Problem 6: Set up variable X (For features) and y from the input data.
X = english_young['WrittenFrequency']
y = english_young['RTlexdec']

# Problem 7: split this into test and train subsets, with 10% of the data in test.

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1, random_state=42)
X_train = X_train.to_numpy().reshape(-1, 1)
X_test = X_test.to_numpy().reshape(-1, 1)

X_plot = np.linspace(0, 10,5000).reshape(-1, 1)
plt.scatter(X_train, y_train, color='red')

print("Model class: " + "Linear Regression")
for degree in [1,2,3,4,5,6,7,20,25]:
  # Problem 8: fit a polynomial regression model of each degree above

  model = make_pipeline(PolynomialFeatures(degree), LinearRegression())
  model.fit(X_train, y_train)

  print("\tDegree " + str(degree) +"\n\t\tTrain R^2: "+ str(model.score(X_train,y_train)))
  print("\t\tTest R^2: "+ str(model.score(X_test,y_test)))

# Problem 9: use plt.plot to add a line to the plot for this model using the X_plot points.

  plt.plot(X_plot, model.predict(X_plot), label=f'Degree {degree}')
  plt.legend(bbox_to_anchor=(1.35, 0), loc='lower right')
  plt.title("Polynomial Regression")
```

```
Model class: Linear Regression
        Degree 1
                Train R^2: 0.42403927186408297
                Test R^2: 0.3210939614227186
        Degree 2
                Train R^2: 0.44436967450777265
                Test R^2: 0.320109824512682
```

```
Degree 3
        Train R^2: 0.4619137563988218
        Test R^2: 0.331296578467405
Degree 4
        Train R^2: 0.4705198262751925
        Test R^2: 0.3401989706400178
Degree 5
        Train R^2: 0.47052035215155297
        Test R^2: 0.3401653611450798
Degree 6
        Train R^2: 0.47171929416375336
        Test R^2: 0.3413840540459021
Degree 7
        Train R^2: 0.4717940795761968
        Test R^2: 0.3420747488899549
Degree 20
        Train R^2: 0.46467163960842883
        Test R^2: 0.3316374446379293
Degree 25
        Train R^2: 0.33935696882297506
        Test R^2: 0.2607235305449904
```



Question 6: Which degree polynomial provided the best fit to this dataset? What does this say about the relationship between frequency and lexical decision times?

Q6: put your answer here (please keep it brief, 2-4 sentences)

Degree 7 polynomial with a test R^2 value of 0.3421 provides the best fit to this dataset which indicates that the behaviour of the relationship between frequency and lexical decision times is better represented by a polynomial model than a linear one. As we increase the degrees, we see a drop in the test R^2 score which is probably because the model is overfitting and therefore the model is not generalizing well to unseen data.

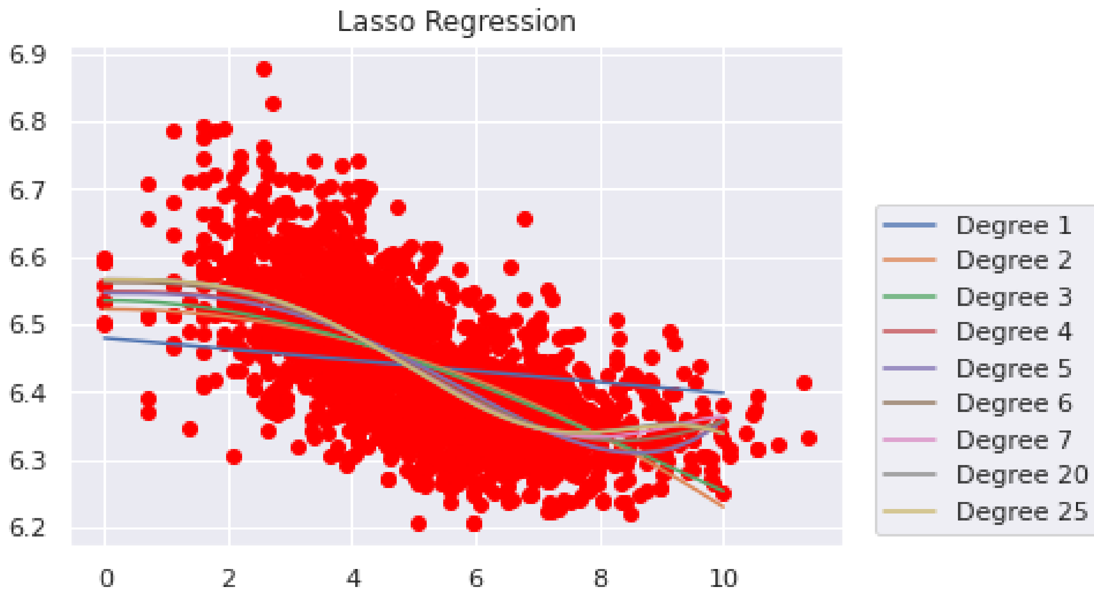
Problem 10

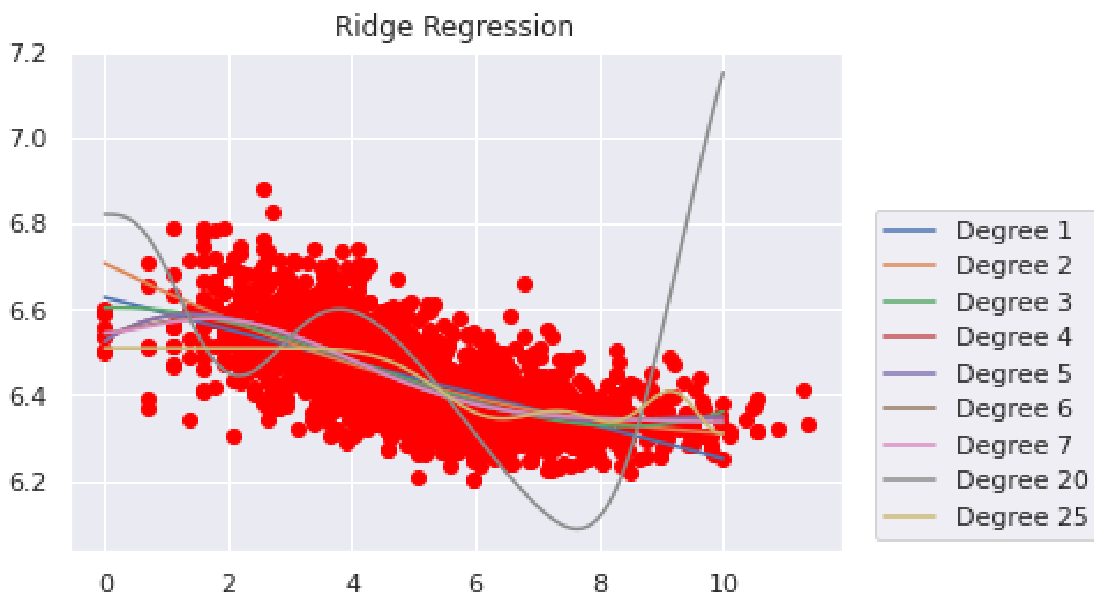
[12]:
```python
# Problem 10: Repeat the above analyses using Lasso and Ridge regression.

from sklearn.linear_model import Lasso, Ridge
import warnings
warnings.filterwarnings("ignore")

degrees = [1, 2, 3, 4, 5, 6, 7, 20, 25]

# iterate over the specified degrees

plt.figure()
plt.scatter(X_train, y_train, color='red')
print("Model class: " + "Lasso Model")
for degree in degrees:
    lasso_model = make_pipeline(PolynomialFeatures(degree), Lasso(alpha=0.1))
    lasso_model.fit(X_train, y_train)
    print("\tDegree " + str(degree) +"\n\t\tTrain R^2: "+ str(lasso_model.
 ↪score(X_train,y_train)))
    print("\t\tTest R^2: "+ str(lasso_model.score(X_test,y_test)))

    plt.plot(X_plot, lasso_model.predict(X_plot), label=f'Degree {degree}')
    plt.legend(bbox_to_anchor=(1.35, 0), loc='lower right')
    plt.title("Lasso Regression")

plt.figure()
plt.scatter(X_train, y_train, color='red')
print("Model class: " + "Ridge Model")
for degree in degrees:
    ridge_model = make_pipeline(PolynomialFeatures(degree), Ridge(alpha=0.1))
    ridge_model.fit(X_train, y_train)
    print("\tDegree " + str(degree) +"\n\t\tTrain R^2: "+ str(ridge_model.
 ↪score(X_train,y_train)))
    print("\t\tTest R^2: "+ str(ridge_model.score(X_test,y_test)))

    plt.plot(X_plot, ridge_model.predict(X_plot), label=f'Degree {degree}')
    plt.legend(bbox_to_anchor=(1.35, 0), loc='lower right')
    plt.title("Ridge Regression")
```

```
Model class: Lasso Model
        Degree 1
                Train R^2: 0.1640863833005869
                Test R^2: 0.140374561077255
```

```
        Degree 2
                Train R^2: 0.3577558927706066
                Test R^2: 0.2956475946184861
        Degree 3
                Train R^2: 0.3875505035726755
                Test R^2: 0.3122754242710505
        Degree 4
                Train R^2: 0.43658956083706735
                Test R^2: 0.3238478720403666
        Degree 5
                Train R^2: 0.43384600081051006
                Test R^2: 0.32239640723787344
        Degree 6
                Train R^2: 0.4567294790154266
                Test R^2: 0.3360512216603645
        Degree 7
                Train R^2: 0.46043645458936255
                Test R^2: 0.33605746582490514
        Degree 20
                Train R^2: 0.4630872307010764
                Test R^2: 0.3370948008069352
        Degree 25
                Train R^2: 0.46327434088411534
                Test R^2: 0.3376391038860217
Model class: Ridge Model
        Degree 1
                Train R^2: 0.42403927177707446
                Test R^2: 0.32109548914409247
        Degree 2
                Train R^2: 0.4443696686169212
                Test R^2: 0.32012250408589427
        Degree 3
                Train R^2: 0.46191375594422857
                Test R^2: 0.33129577679649924
        Degree 4
                Train R^2: 0.4705194293656332
                Test R^2: 0.3401570442258436
        Degree 5
                Train R^2: 0.4705192301641664
                Test R^2: 0.3400840989856295
        Degree 6
                Train R^2: 0.47171928201977187
                Test R^2: 0.3413768027745354
        Degree 7
                Train R^2: 0.47179259884539226
                Test R^2: 0.3420068519592019
        Degree 20
                Train R^2: -1.6471277997487035
```

```
        Test R^2: -1.3666402128135426
Degree 25
        Train R^2: 0.399195517330734
        Test R^2: 0.2732587410941144
```





Question 7: What did you find using these regularization techniques? How were they similar or different to eachother and the results above? Why?

Q7: put your answer here (please keep it brief, 4-6 sentences)

In comparison to the polynomial regression model, both of these regularization techniques do not yield significantly improved test R^2 scores. For degrees upto 3 we do see that test and train scores are closer for Lasso, indicating some overfitting was addressed. Higher degrees show very similar results to the polynomial model above. For Ridge model, we don't see any considerable improvements. In fact, at degree 20 we encounter a negative train score which is not what we want.

There are a few reasons why these models can behave this way. We should be looking at tuning the hyperparameters such as the learning rate. Perhaps the learning rate of 0.1 is too big which can also explain why the behaviour for Ridge model at 20 degrees. Having more data can also greatly improve the model. Performing a grid search with cross-validation can help us find the optimal parameters for both models.

Now we will look at aniother dataset available here.

This dataset is about morphological *regularity*—the property of whether words marke certain information like tense using regular endings (e.g., *walk*/*walked*) or irregular processes (e.g., *sing*/*sang*). The dataset consists of a Dutch verbs and is described in the following paper.

Tabak, W. M., Schreuder, R., and Baayen, R. H. (2005). Lexical statistics and lexical processing: Semantic density, information complexity, sex, and irregularity in Dutch. In Kesper, S. and Reis, M., editors, Linguistic Evidence — Empirical, Theoretical, and Computational Perspectives, pages 529–555. Mouton de Gruyter, Berlin, Germany.

Figure 1 in the paper displays the correlations between various factors (features) and predictability.

Problem 11

[14]:
```
#Problem 11: Read in the regularity dataset and familiarize yourself with it.

regularity = pd.read_csv('/content/drive/My Drive/regularity.csv')
display(regularity)
```

```
     Unnamed: 0     Verb  WrittenFrequency  FamilySize  LengthInLetters  \
0             1   stelen          1.609438    2.564949                5
1             2   tollen          5.411646    2.397895                3
2             3  blijken          9.883183    1.791759                5
3             4  gloeien          6.908755    2.079442                5
4             5   kakken          3.784190    2.079442                3
..          …       …               …          …              …
695         696   volgen          9.986035    4.043051                4
696         697  ploffen          5.533389    2.302585                4
697         698   stelen          7.391415    2.564949                5
698         699    jagen          7.682022    4.007333                4
699         700    jagen          7.682022    4.007333                4

     VerbalSynsets  MeanBigramFrequency  NcountStem Regularity  \
0                1                14.47           9    regular
1                1                13.61          26    regular
2                1                13.84           6  irregular
```

```
3                3              12.95           4    regular
4                1              13.48          24    regular
..             …               …          …         …
695              7              13.44           4    regular
696              4              12.82           5    regular
697              1              14.47           9  irregular
698              5              13.32          11    regular
699              5              13.32          11  irregular

     InflectionalEntropy Auxiliary  Valency   NVratio  WrittenSpokenRatio
0                1.00000    hebben        3  4.366913           -2.302585
1                2.48640   zijnheb        3 -0.078927           -2.351375
2                2.00458      zijn        4 -3.192341           -1.526056
3                2.56009    hebben        5 -6.908755            1.791759
4                1.99257    hebben        2 -0.526093           -1.386294
..                   …         …        …         …                   …
695              2.61010   zijnheb        5 -9.986035            0.543615
696              2.38985   zijnheb        4 -1.011601           -0.154151
697              1.77808    hebben        3 -1.415064           -0.105361
698              2.57416    hebben        7 -7.682022            1.734601
699              2.57416    hebben        7 -7.682022            1.734601

[700 rows x 14 columns]
```

Question 8: Briefly describe what each of the 5 factors excluding inflectional entropy, log argument structures, and log N-count means and what relationship it shows to regularity.

Q8: put your answer here. (no more than 1-2 sentences per factor)

In this section, we will focus on the relationship between `FamilySize` and `Regularity`. Unlike in the last dataset, regularity is not a continuous value, but rather a binary variable. Thus we will need to use a classification model to examine it.

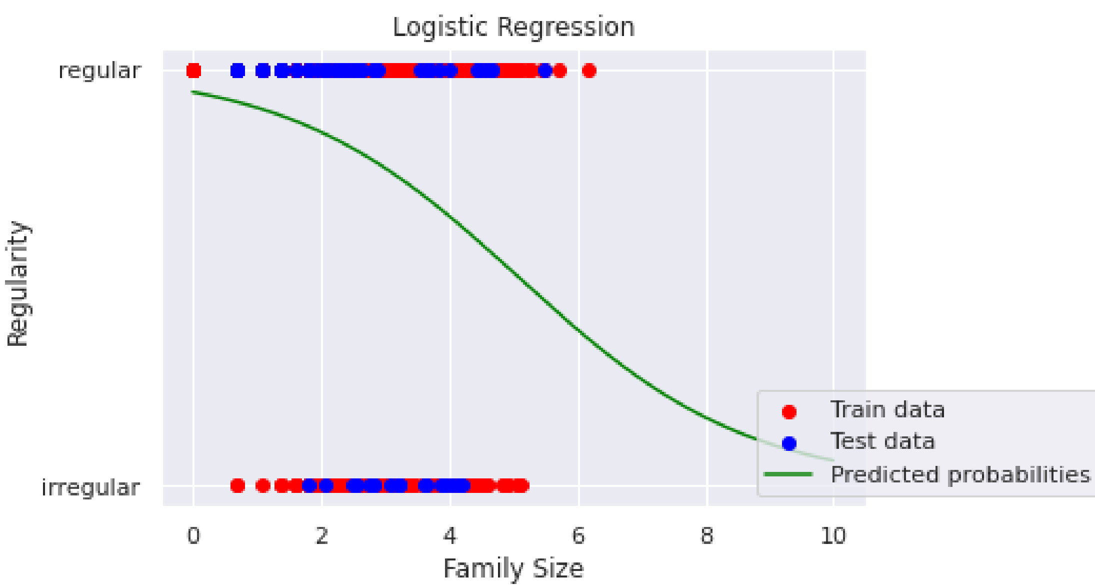
For this pupose, we will use logistic regression.

Problems 12-15

[15]:
```python
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

#Problem 12: set up X as FamilySize, y as Regularity in
#preparation to use them to fit a logistic regression model.

X = regularity['FamilySize']
y = regularity['Regularity']

#Problem 13: split X, and y into train and test with 10% test split. And also
↪create
# and X_plot variable for plotting.
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1,␣
 ↪random_state=42)
X_train = X_train.to_numpy().reshape(-1, 1)
X_test = X_test.to_numpy().reshape(-1, 1)
X_plot = np.linspace(0, 10, 5000).reshape(-1, 1)

#Problem 14: fit a logistic regression model on this data

lr_model = LogisticRegression()
lr_model.fit(X_train,y_train)

display([accuracy_score(y_train, lr_model.
 ↪predict(X_train)),accuracy_score(y_test, lr_model.predict(X_test))])

#Problem 15: using plt.plot make a scatter plot of the data along with the␣
 ↪predicted
# probability of regularity as a function of family size. you will find the␣
 ↪"predict_proba" function helpful.

probas = lr_model.predict_proba(X_plot)[:, 1]
plt.scatter(X_train, y_train, color='red', label='Train data')
plt.scatter(X_test, y_test, color='blue', label='Test data')
plt.plot(X_plot, probas, color='green', label='Predicted probabilities')
plt.legend(bbox_to_anchor=(1.35, 0), loc='lower right')
plt.xlabel('Family Size')
plt.ylabel('Regularity')
plt.title('Logistic Regression')
plt.show()
```

```
[0.765079365079365, 0.7285714285714285]
```



Question 9: What do you see in the fit to the logistic regression? How does the probability of being regular vary with morphological family size?

Q9: put your answer here. (no more than 2-4 sentences)

The probability of being regular decreases as morphological family size increases. This suggests that irregular forms tend to be more common in larger morphological families than smaller families.

Question 10: What does the relationship on Figure 1 of the paper look like? Does this match your analysis above?

Q10: put your answer here. (no more than 2-4 sentences)

The relationship on figure 1 of the paper does not match the analysis above exactly. In the paper, figure 1 shows that probability of regularity does decrease but then goes back up so it is not clear if a higher family sizes will indeed result in lower probabilities of being regular.

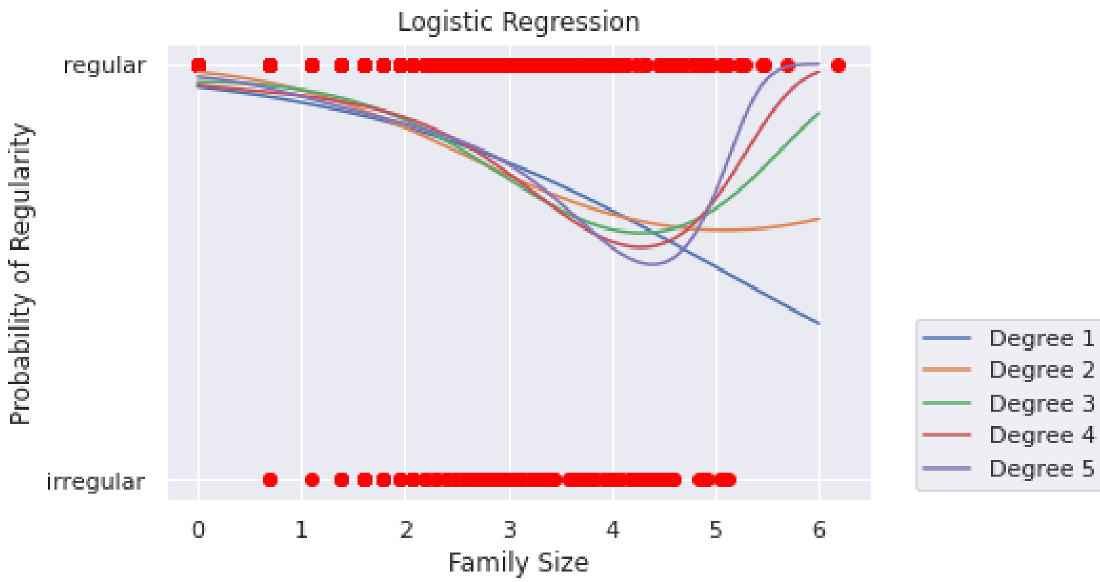
In class, we saw an example of a regression problem in which the curve was best fit by a polynomial. We can, of course, use polynomial relationships in a classification model as well. Let's take a similar approach to the analysis that we just did, except first transforming our input features polynomially.

Problems 16-19

[16]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import PolynomialFeatures
from sklearn.metrics import accuracy_score
import  scipy

#Problem 16: Set up X, y, test and train as above.

X = regularity['FamilySize']
y = regularity['Regularity']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1,␣
 ↪random_state=42)
X_train = X_train.to_numpy().reshape(-1, 1)
X_test = X_test.to_numpy().reshape(-1, 1)

X_plot = np.linspace(0, 6,5000).reshape(-1, 1)


#Problem 17: Plot the data points using plt.scatter.

plt.scatter(X_train,y_train,color='red')

print("Model class: " + "Logistic Regression")
for degree in [1,2,3,4,5]:
```

```python
#Problem 18: set up a logistic regression model with polynomial features of␣
↪degree and train it.

lr_model = make_pipeline(PolynomialFeatures(degree), LogisticRegression())
lr_model.fit(X_train, y_train)

print("\tDegree " + str(degree) +"\n\t\tTrain R^2: "+␣
↪str(accuracy_score(y_train, lr_model.predict(X_train))))
print("\t\tTest R^2: "+ str(accuracy_score(y_test, lr_model.predict(X_test))))

#Problem 19: Plot the resulting predicted probability line on the plot.

# Put your answer here

probs = lr_model.predict_proba(X_plot)[:, 1]
plt.plot(X_plot, probs, label=f'Degree {degree}')

plt.legend(bbox_to_anchor=(1.35, 0), loc='lower right')
plt.title("Logistic Regression")
plt.xlabel('Family Size')
plt.ylabel('Probability of Regularity')
```

```
Model class: Logistic Regression
        Degree 1
                Train R^2: 0.765079365079365
                Test R^2: 0.7285714285714285
        Degree 2
                Train R^2: 0.7761904761904762
                Test R^2: 0.7428571428571429
        Degree 3
                Train R^2: 0.7761904761904762
                Test R^2: 0.7428571428571429
        Degree 4
                Train R^2: 0.7761904761904762
                Test R^2: 0.7428571428571429
        Degree 5
                Train R^2: 0.7761904761904762
                Test R^2: 0.7428571428571429
```



Question 11: What do you see in the fit of these polynomial features? What is the relationship in plain English between the family size variable and the probability of being regular?

Q11: put your answer here. (no more than 2-4 sentences)

Here we can see that the fit of these polynomial features is very close to the fit shown in figure 1 in the paper. We see the curves decrease with family size and then present slightlt differing behaviour past a family size of about 5. If we look at the best fit degree models, which in this case models of degrees 2, 3, 4, and 5 present the same test R^2 scores, the probability of regularity increases after about a family size of 5.

In other words, the probability of being regular decreases as family size increases to a certain point after which the probability returns to an upward trend.

Question 12: Speculate as to why this relationship might hold theoretically?

Q12: put your answer here. (no more than 2-4 sentences)

Since the probability of regularity decreases until a certain size and then goes back up, it suggests that there is a non-linear relationship between family size and regularity. Its possible that the irregular forms of words with higher morphological family sizes have becomes more standardized over time which has lead to an increase in regularity for those words. It's also important to explore other factors that could be contributing to this relationship to get a full understanding of their relationship.

1 To Submit

To submit, name this notebook `YOUR_STUDENT_ID_Assignment_4.ipynb`, then convert this `.ipynb` file to a `.pdf` (e.g., using the following instructions) and upload the PDF to the Gradescope assignment “Assignment 4 – Psycholinguistic data and regression”.

(Note: `Print > Save as PDF` **will not work** because it will not display your figures correctly.)

You can convert the notebook to a PDF using the following instructions.

2 Converting this notebook to a PDF

1. Make sure you have renamed the notebook, e.g. `000000000_Assignment_4.ipynb` where `000000000` is your student ID.
2. Make sure to save the notebook (`ctrl/cmd + s`).

2. Make sure Google Drive is mounted (it likely already is from the first question).

[17]:
```
from google.colab import drive
drive.mount('/content/drive/')
!ls "/content/drive/MyDrive/Colab Notebooks/"
```

```
Drive already mounted at /content/drive/; to attempt to forcibly remount, call
drive.mount("/content/drive/", force_remount=True).
 260925886_260921696_Assignment_2.ipynb   'Copy of Question 6 2nd (1)'
 260925886_Assignment_4.ipynb             'Copy of Question 6 2nd (2)'
 345asgn3                                 'Copy of Question 6 2nd (3)'
'Copy of Copy of MiniProject2.ipynb'      'Copy of Question 6 PT'
'Copy of Copy of Untitled0.ipynb'         'Copy of Untitled0.ipynb'
'Copy of MiniProject2.ipynb'              'Question 6 2nd'
'Copy of NL2DS-W2023-Assignment-1.ipynb'  'Question 6 PT.ipynb'
'Copy of Question 6 2nd'                   Untitled0.ipynb
```

3. Install packages for converting .ipynb to .pdf

[18]:
```
!apt-get -q install texlive-xetex texlive-fonts-recommended
  texlive-plain-generic
```

```
Reading package lists…
Building dependency tree…
Reading state information…
The following additional packages will be installed:
  dvisvgm fonts-droid-fallback fonts-lato fonts-lmodern fonts-noto-mono
  fonts-texgyre fonts-urw-base35 javascript-common libapache-pom-java
  libcommons-logging-java libcommons-parent-java libfontbox-java libgs9
  libgs9-common libidn11 libijs-0.35 libjbig2dec0 libjs-jquery libkpathsea6
  libpdfbox-java libptexenc1 libruby2.7 libsynctex2 libteckit0 libtexlua53
  libtexluajit2 libzzip-0-13 lmodern poppler-data preview-latex-style rake
  ruby ruby-minitest ruby-net-telnet ruby-power-assert ruby-test-unit
  ruby-xmlrpc ruby2.7 rubygems-integration t1utils teckit tex-common tex-gyre
  texlive-base texlive-binaries texlive-latex-base texlive-latex-extra
  texlive-latex-recommended texlive-pictures tipa xfonts-encodings
  xfonts-utils
Suggested packages:
  fonts-noto fonts-freefont-otf | fonts-freefont-ttf apache2 | lighttpd
```

```
  | httpd libavalon-framework-java libcommons-logging-java-doc
  libexcalibur-logkit-java liblog4j1.2-java poppler-utils ghostscript
  fonts-japanese-mincho | fonts-ipafont-mincho fonts-japanese-gothic
  | fonts-ipafont-gothic fonts-arphic-ukai fonts-arphic-uming fonts-nanum ri
  ruby-dev bundler debhelper gv | postscript-viewer perl-tk xpdf | pdf-viewer
  xzdec texlive-fonts-recommended-doc texlive-latex-base-doc python3-pygments
  icc-profiles libfile-which-perl libspreadsheet-parseexcel-perl
  texlive-latex-extra-doc texlive-latex-recommended-doc texlive-luatex
  texlive-pstricks dot2tex prerex ruby-tcltk | libtcltk-ruby
  texlive-pictures-doc vprerex
The following NEW packages will be installed:
  dvisvgm fonts-droid-fallback fonts-lato fonts-lmodern fonts-noto-mono
  fonts-texgyre fonts-urw-base35 javascript-common libapache-pom-java
  libcommons-logging-java libcommons-parent-java libfontbox-java libgs9
  libgs9-common libidn11 libijs-0.35 libjbig2dec0 libjs-jquery libkpathsea6
  libpdfbox-java libptexenc1 libruby2.7 libsynctex2 libteckit0 libtexlua53
  libtexluajit2 libzzip-0-13 lmodern poppler-data preview-latex-style rake
  ruby ruby-minitest ruby-net-telnet ruby-power-assert ruby-test-unit
  ruby-xmlrpc ruby2.7 rubygems-integration t1utils teckit tex-common tex-gyre
  texlive-base texlive-binaries texlive-fonts-recommended texlive-latex-base
  texlive-latex-extra texlive-latex-recommended texlive-pictures
  texlive-plain-generic texlive-xetex tipa xfonts-encodings xfonts-utils
0 upgraded, 55 newly installed, 0 to remove and 23 not upgraded.
Need to get 169 MB of archives.
After this operation, 536 MB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu focal/main amd64 fonts-droid-fallback all
1:6.0.1r16-1.1 [1,805 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal/main amd64 fonts-lato all 2.0-2
[2,698 kB]
Get:3 http://archive.ubuntu.com/ubuntu focal/main amd64 poppler-data all 0.4.9-2
[1,475 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal/universe amd64 tex-common all 6.13
[32.7 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal/main amd64 fonts-urw-base35 all
20170801.1-3 [6,333 kB]
Get:6 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 libgs9-common
all 9.50~dfsg-5ubuntu4.6 [681 kB]
Get:7 http://archive.ubuntu.com/ubuntu focal/main amd64 libidn11 amd64
1.33-2.2ubuntu2 [46.2 kB]
Get:8 http://archive.ubuntu.com/ubuntu focal/main amd64 libijs-0.35 amd64
0.35-15 [15.7 kB]
Get:9 http://archive.ubuntu.com/ubuntu focal/main amd64 libjbig2dec0 amd64
0.18-1ubuntu1 [60.0 kB]
Get:10 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 libgs9 amd64
9.50~dfsg-5ubuntu4.6 [2,173 kB]
Get:11 http://archive.ubuntu.com/ubuntu focal/main amd64 libkpathsea6 amd64
2019.20190605.51237-3build2 [57.0 kB]
Get:12 http://archive.ubuntu.com/ubuntu focal/universe amd64 dvisvgm amd64
```

```
2.8.1-1build1 [1,048 kB]
Get:13 http://archive.ubuntu.com/ubuntu focal/universe amd64 fonts-lmodern all
2.004.5-6 [4,532 kB]
Get:14 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 fonts-noto-mono
all 20200323-1build1~ubuntu20.04.1 [80.6 kB]
Get:15 http://archive.ubuntu.com/ubuntu focal/universe amd64 fonts-texgyre all
20180621-3 [10.2 MB]
Get:16 http://archive.ubuntu.com/ubuntu focal/main amd64 javascript-common all
11 [6,066 B]
Get:17 http://archive.ubuntu.com/ubuntu focal/universe amd64 libapache-pom-java
all 18-1 [4,720 B]
Get:18 http://archive.ubuntu.com/ubuntu focal/universe amd64 libcommons-parent-
java all 43-1 [10.8 kB]
Get:19 http://archive.ubuntu.com/ubuntu focal/universe amd64 libcommons-logging-
java all 1.2-2 [60.3 kB]
Get:20 http://archive.ubuntu.com/ubuntu focal/main amd64 libjs-jquery all
3.3.1~dfsg-3 [329 kB]
Get:21 http://archive.ubuntu.com/ubuntu focal/main amd64 libptexenc1 amd64
2019.20190605.51237-3build2 [35.5 kB]
Get:22 http://archive.ubuntu.com/ubuntu focal/main amd64 rubygems-integration
all 1.16 [5,092 B]
Get:23 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 ruby2.7 amd64
2.7.0-5ubuntu1.7 [95.6 kB]
Get:24 http://archive.ubuntu.com/ubuntu focal/main amd64 ruby amd64 1:2.7+1
[5,412 B]
Get:25 http://archive.ubuntu.com/ubuntu focal/main amd64 rake all 13.0.1-4 [61.6
kB]
Get:26 http://archive.ubuntu.com/ubuntu focal/main amd64 ruby-minitest all
5.13.0-1 [40.9 kB]
Get:27 http://archive.ubuntu.com/ubuntu focal/main amd64 ruby-net-telnet all
0.1.1-2 [12.6 kB]
Get:28 http://archive.ubuntu.com/ubuntu focal/main amd64 ruby-power-assert all
1.1.7-1 [11.4 kB]
Get:29 http://archive.ubuntu.com/ubuntu focal/main amd64 ruby-test-unit all
3.3.5-1 [73.2 kB]
Get:30 http://archive.ubuntu.com/ubuntu focal/main amd64 ruby-xmlrpc all 0.3.0-2
[23.8 kB]
Get:31 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 libruby2.7
amd64 2.7.0-5ubuntu1.7 [3,533 kB]
Get:32 http://archive.ubuntu.com/ubuntu focal/main amd64 libsynctex2 amd64
2019.20190605.51237-3build2 [55.0 kB]
Get:33 http://archive.ubuntu.com/ubuntu focal/universe amd64 libteckit0 amd64
2.5.8+ds2-5ubuntu2 [320 kB]
Get:34 http://archive.ubuntu.com/ubuntu focal/main amd64 libtexlua53 amd64
2019.20190605.51237-3build2 [105 kB]
Get:35 http://archive.ubuntu.com/ubuntu focal/main amd64 libtexluajit2 amd64
2019.20190605.51237-3build2 [235 kB]
Get:36 http://archive.ubuntu.com/ubuntu focal/universe amd64 libzzip-0-13 amd64
```

```
0.13.62-3.2ubuntu1 [26.2 kB]
Get:37 http://archive.ubuntu.com/ubuntu focal/main amd64 xfonts-encodings all
1:1.0.5-0ubuntu1 [573 kB]
Get:38 http://archive.ubuntu.com/ubuntu focal/main amd64 xfonts-utils amd64
1:7.7+6 [91.5 kB]
Get:39 http://archive.ubuntu.com/ubuntu focal/universe amd64 lmodern all
2.004.5-6 [9,474 kB]
Get:40 http://archive.ubuntu.com/ubuntu focal/universe amd64 preview-latex-style
all 11.91-2ubuntu2 [184 kB]
Get:41 http://archive.ubuntu.com/ubuntu focal/main amd64 t1utils amd64 1.41-3
[56.1 kB]
Get:42 http://archive.ubuntu.com/ubuntu focal/universe amd64 teckit amd64
2.5.8+ds2-5ubuntu2 [687 kB]
Get:43 http://archive.ubuntu.com/ubuntu focal/universe amd64 tex-gyre all
20180621-3 [6,209 kB]
Get:44 http://archive.ubuntu.com/ubuntu focal/universe amd64 texlive-binaries
amd64 2019.20190605.51237-3build2 [8,041 kB]
Get:45 http://archive.ubuntu.com/ubuntu focal/universe amd64 texlive-base all
2019.20200218-1 [20.8 MB]
Get:46 http://archive.ubuntu.com/ubuntu focal/universe amd64 texlive-fonts-
recommended all 2019.20200218-1 [4,972 kB]
Get:47 http://archive.ubuntu.com/ubuntu focal/universe amd64 texlive-latex-base
all 2019.20200218-1 [990 kB]
Get:48 http://archive.ubuntu.com/ubuntu focal/universe amd64 libfontbox-java all
1:1.8.16-2 [207 kB]
Get:49 http://archive.ubuntu.com/ubuntu focal/universe amd64 libpdfbox-java all
1:1.8.16-2 [5,199 kB]
Get:50 http://archive.ubuntu.com/ubuntu focal/universe amd64 texlive-latex-
recommended all 2019.20200218-1 [15.7 MB]
Get:51 http://archive.ubuntu.com/ubuntu focal/universe amd64 texlive-pictures
all 2019.20200218-1 [4,492 kB]
Get:52 http://archive.ubuntu.com/ubuntu focal/universe amd64 texlive-latex-extra
all 2019.202000218-1 [12.5 MB]
Get:53 http://archive.ubuntu.com/ubuntu focal/universe amd64 texlive-plain-
generic all 2019.202000218-1 [24.6 MB]
Get:54 http://archive.ubuntu.com/ubuntu focal/universe amd64 tipa all 2:1.3-20
[2,978 kB]
Get:55 http://archive.ubuntu.com/ubuntu focal/universe amd64 texlive-xetex all
2019.20200218-1 [14.6 MB]
Fetched 169 MB in 4s (46.2 MB/s)
Extracting templates from packages: 100%
Preconfiguring packages …
Selecting previously unselected package fonts-droid-fallback.
(Reading database … 128276 files and directories currently installed.)
Preparing to unpack …/00-fonts-droid-fallback_1%3a6.0.1r16-1.1_all.deb …
Unpacking fonts-droid-fallback (1:6.0.1r16-1.1) …
Selecting previously unselected package fonts-lato.
Preparing to unpack …/01-fonts-lato_2.0-2_all.deb …
```

```
Unpacking fonts-lato (2.0-2) …
Selecting previously unselected package poppler-data.
Preparing to unpack …/02-poppler-data_0.4.9-2_all.deb …
Unpacking poppler-data (0.4.9-2) …
Selecting previously unselected package tex-common.
Preparing to unpack …/03-tex-common_6.13_all.deb …
Unpacking tex-common (6.13) …
Selecting previously unselected package fonts-urw-base35.
Preparing to unpack …/04-fonts-urw-base35_20170801.1-3_all.deb …
Unpacking fonts-urw-base35 (20170801.1-3) …
Selecting previously unselected package libgs9-common.
Preparing to unpack …/05-libgs9-common_9.50~dfsg-5ubuntu4.6_all.deb …
Unpacking libgs9-common (9.50~dfsg-5ubuntu4.6) …
Selecting previously unselected package libidn11:amd64.
Preparing to unpack …/06-libidn11_1.33-2.2ubuntu2_amd64.deb …
Unpacking libidn11:amd64 (1.33-2.2ubuntu2) …
Selecting previously unselected package libijs-0.35:amd64.
Preparing to unpack …/07-libijs-0.35_0.35-15_amd64.deb …
Unpacking libijs-0.35:amd64 (0.35-15) …
Selecting previously unselected package libjbig2dec0:amd64.
Preparing to unpack …/08-libjbig2dec0_0.18-1ubuntu1_amd64.deb …
Unpacking libjbig2dec0:amd64 (0.18-1ubuntu1) …
Selecting previously unselected package libgs9:amd64.
Preparing to unpack …/09-libgs9_9.50~dfsg-5ubuntu4.6_amd64.deb …
Unpacking libgs9:amd64 (9.50~dfsg-5ubuntu4.6) …
Selecting previously unselected package libkpathsea6:amd64.
Preparing to unpack …/10-libkpathsea6_2019.20190605.51237-3build2_amd64.deb
…
Unpacking libkpathsea6:amd64 (2019.20190605.51237-3build2) …
Selecting previously unselected package dvisvgm.
Preparing to unpack …/11-dvisvgm_2.8.1-1build1_amd64.deb …
Unpacking dvisvgm (2.8.1-1build1) …
Selecting previously unselected package fonts-lmodern.
Preparing to unpack …/12-fonts-lmodern_2.004.5-6_all.deb …
Unpacking fonts-lmodern (2.004.5-6) …
Selecting previously unselected package fonts-noto-mono.
Preparing to unpack …/13-fonts-noto-
mono_20200323-1build1~ubuntu20.04.1_all.deb …
Unpacking fonts-noto-mono (20200323-1build1~ubuntu20.04.1) …
Selecting previously unselected package fonts-texgyre.
Preparing to unpack …/14-fonts-texgyre_20180621-3_all.deb …
Unpacking fonts-texgyre (20180621-3) …
Selecting previously unselected package javascript-common.
Preparing to unpack …/15-javascript-common_11_all.deb …
Unpacking javascript-common (11) …
Selecting previously unselected package libapache-pom-java.
Preparing to unpack …/16-libapache-pom-java_18-1_all.deb …
Unpacking libapache-pom-java (18-1) …
```

```
Selecting previously unselected package libcommons-parent-java.
Preparing to unpack …/17-libcommons-parent-java_43-1_all.deb …
Unpacking libcommons-parent-java (43-1) …
Selecting previously unselected package libcommons-logging-java.
Preparing to unpack …/18-libcommons-logging-java_1.2-2_all.deb …
Unpacking libcommons-logging-java (1.2-2) …
Selecting previously unselected package libjs-jquery.
Preparing to unpack …/19-libjs-jquery_3.3.1~dfsg-3_all.deb …
Unpacking libjs-jquery (3.3.1~dfsg-3) …
Selecting previously unselected package libptexenc1:amd64.
Preparing to unpack …/20-libptexenc1_2019.20190605.51237-3build2_amd64.deb …
Unpacking libptexenc1:amd64 (2019.20190605.51237-3build2) …
Selecting previously unselected package rubygems-integration.
Preparing to unpack …/21-rubygems-integration_1.16_all.deb …
Unpacking rubygems-integration (1.16) …
Selecting previously unselected package ruby2.7.
Preparing to unpack …/22-ruby2.7_2.7.0-5ubuntu1.7_amd64.deb …
Unpacking ruby2.7 (2.7.0-5ubuntu1.7) …
Selecting previously unselected package ruby.
Preparing to unpack …/23-ruby_1%3a2.7+1_amd64.deb …
Unpacking ruby (1:2.7+1) …
Selecting previously unselected package rake.
Preparing to unpack …/24-rake_13.0.1-4_all.deb …
Unpacking rake (13.0.1-4) …
Selecting previously unselected package ruby-minitest.
Preparing to unpack …/25-ruby-minitest_5.13.0-1_all.deb …
Unpacking ruby-minitest (5.13.0-1) …
Selecting previously unselected package ruby-net-telnet.
Preparing to unpack …/26-ruby-net-telnet_0.1.1-2_all.deb …
Unpacking ruby-net-telnet (0.1.1-2) …
Selecting previously unselected package ruby-power-assert.
Preparing to unpack …/27-ruby-power-assert_1.1.7-1_all.deb …
Unpacking ruby-power-assert (1.1.7-1) …
Selecting previously unselected package ruby-test-unit.
Preparing to unpack …/28-ruby-test-unit_3.3.5-1_all.deb …
Unpacking ruby-test-unit (3.3.5-1) …
Selecting previously unselected package ruby-xmlrpc.
Preparing to unpack …/29-ruby-xmlrpc_0.3.0-2_all.deb …
Unpacking ruby-xmlrpc (0.3.0-2) …
Selecting previously unselected package libruby2.7:amd64.
Preparing to unpack …/30-libruby2.7_2.7.0-5ubuntu1.7_amd64.deb …
Unpacking libruby2.7:amd64 (2.7.0-5ubuntu1.7) …
Selecting previously unselected package libsynctex2:amd64.
Preparing to unpack …/31-libsynctex2_2019.20190605.51237-3build2_amd64.deb …
Unpacking libsynctex2:amd64 (2019.20190605.51237-3build2) …
Selecting previously unselected package libteckit0:amd64.
Preparing to unpack …/32-libteckit0_2.5.8+ds2-5ubuntu2_amd64.deb …
Unpacking libteckit0:amd64 (2.5.8+ds2-5ubuntu2) …
```

```
Selecting previously unselected package libtexlua53:amd64.
Preparing to unpack …/33-libtexlua53_2019.20190605.51237-3build2_amd64.deb …
Unpacking libtexlua53:amd64 (2019.20190605.51237-3build2) …
Selecting previously unselected package libtexluajit2:amd64.
Preparing to unpack …/34-libtexluajit2_2019.20190605.51237-3build2_amd64.deb
…
Unpacking libtexluajit2:amd64 (2019.20190605.51237-3build2) …
Selecting previously unselected package libzzip-0-13:amd64.
Preparing to unpack …/35-libzzip-0-13_0.13.62-3.2ubuntu1_amd64.deb …
Unpacking libzzip-0-13:amd64 (0.13.62-3.2ubuntu1) …
Selecting previously unselected package xfonts-encodings.
Preparing to unpack …/36-xfonts-encodings_1%3a1.0.5-0ubuntu1_all.deb …
Unpacking xfonts-encodings (1:1.0.5-0ubuntu1) …
Selecting previously unselected package xfonts-utils.
Preparing to unpack …/37-xfonts-utils_1%3a7.7+6_amd64.deb …
Unpacking xfonts-utils (1:7.7+6) …
Selecting previously unselected package lmodern.
Preparing to unpack …/38-lmodern_2.004.5-6_all.deb …
Unpacking lmodern (2.004.5-6) …
Selecting previously unselected package preview-latex-style.
Preparing to unpack …/39-preview-latex-style_11.91-2ubuntu2_all.deb …
Unpacking preview-latex-style (11.91-2ubuntu2) …
Selecting previously unselected package t1utils.
Preparing to unpack …/40-t1utils_1.41-3_amd64.deb …
Unpacking t1utils (1.41-3) …
Selecting previously unselected package teckit.
Preparing to unpack …/41-teckit_2.5.8+ds2-5ubuntu2_amd64.deb …
Unpacking teckit (2.5.8+ds2-5ubuntu2) …
Selecting previously unselected package tex-gyre.
Preparing to unpack …/42-tex-gyre_20180621-3_all.deb …
Unpacking tex-gyre (20180621-3) …
Selecting previously unselected package texlive-binaries.
Preparing to unpack …/43-texlive-
binaries_2019.20190605.51237-3build2_amd64.deb …
Unpacking texlive-binaries (2019.20190605.51237-3build2) …
Selecting previously unselected package texlive-base.
Preparing to unpack …/44-texlive-base_2019.20200218-1_all.deb …
Unpacking texlive-base (2019.20200218-1) …
Selecting previously unselected package texlive-fonts-recommended.
Preparing to unpack …/45-texlive-fonts-recommended_2019.20200218-1_all.deb …
Unpacking texlive-fonts-recommended (2019.20200218-1) …
Selecting previously unselected package texlive-latex-base.
Preparing to unpack …/46-texlive-latex-base_2019.20200218-1_all.deb …
Unpacking texlive-latex-base (2019.20200218-1) …
Selecting previously unselected package libfontbox-java.
Preparing to unpack …/47-libfontbox-java_1%3a1.8.16-2_all.deb …
Unpacking libfontbox-java (1:1.8.16-2) …
Selecting previously unselected package libpdfbox-java.
```

```
Preparing to unpack …/48-libpdfbox-java_1%3a1.8.16-2_all.deb …
Unpacking libpdfbox-java (1:1.8.16-2) …
Selecting previously unselected package texlive-latex-recommended.
Preparing to unpack …/49-texlive-latex-recommended_2019.20200218-1_all.deb …
Unpacking texlive-latex-recommended (2019.20200218-1) …
Selecting previously unselected package texlive-pictures.
Preparing to unpack …/50-texlive-pictures_2019.20200218-1_all.deb …
Unpacking texlive-pictures (2019.20200218-1) …
Selecting previously unselected package texlive-latex-extra.
Preparing to unpack …/51-texlive-latex-extra_2019.202000218-1_all.deb …
Unpacking texlive-latex-extra (2019.202000218-1) …
Selecting previously unselected package texlive-plain-generic.
Preparing to unpack …/52-texlive-plain-generic_2019.202000218-1_all.deb …
Unpacking texlive-plain-generic (2019.202000218-1) …
Selecting previously unselected package tipa.
Preparing to unpack …/53-tipa_2%3a1.3-20_all.deb …
Unpacking tipa (2:1.3-20) …
Selecting previously unselected package texlive-xetex.
Preparing to unpack …/54-texlive-xetex_2019.20200218-1_all.deb …
Unpacking texlive-xetex (2019.20200218-1) …
Setting up javascript-common (11) …
Setting up fonts-lato (2.0-2) …
Setting up fonts-noto-mono (20200323-1build1~ubuntu20.04.1) …
Setting up ruby-power-assert (1.1.7-1) …
Setting up libtexlua53:amd64 (2019.20190605.51237-3build2) …
Setting up libijs-0.35:amd64 (0.35-15) …
Setting up libtexluajit2:amd64 (2019.20190605.51237-3build2) …
Setting up libfontbox-java (1:1.8.16-2) …
Setting up rubygems-integration (1.16) …
Setting up libzzip-0-13:amd64 (0.13.62-3.2ubuntu1) …
Setting up fonts-urw-base35 (20170801.1-3) …
Setting up poppler-data (0.4.9-2) …
Setting up ruby-minitest (5.13.0-1) …
Setting up tex-common (6.13) …
update-language: texlive-base not installed and configured, doing nothing!
Setting up ruby-test-unit (3.3.5-1) …
Setting up libjbig2dec0:amd64 (0.18-1ubuntu1) …
Setting up libidn11:amd64 (1.33-2.2ubuntu2) …
Setting up libteckit0:amd64 (2.5.8+ds2-5ubuntu2) …
Setting up libapache-pom-java (18-1) …
Setting up ruby-net-telnet (0.1.1-2) …
Setting up xfonts-encodings (1:1.0.5-0ubuntu1) …
Setting up t1utils (1.41-3) …
Setting up fonts-texgyre (20180621-3) …
Setting up libkpathsea6:amd64 (2019.20190605.51237-3build2) …
Setting up fonts-lmodern (2.004.5-6) …
Setting up fonts-droid-fallback (1:6.0.1r16-1.1) …
Setting up libjs-jquery (3.3.1~dfsg-3) …
```

```
Setting up ruby-xmlrpc (0.3.0-2) …
Setting up libsynctex2:amd64 (2019.20190605.51237-3build2) …
Setting up libgs9-common (9.50~dfsg-5ubuntu4.6) …
Setting up teckit (2.5.8+ds2-5ubuntu2) …
Setting up libpdfbox-java (1:1.8.16-2) …
Setting up libgs9:amd64 (9.50~dfsg-5ubuntu4.6) …
Setting up preview-latex-style (11.91-2ubuntu2) …
Setting up libcommons-parent-java (43-1) …
Setting up dvisvgm (2.8.1-1build1) …
Setting up libcommons-logging-java (1.2-2) …
Setting up xfonts-utils (1:7.7+6) …
Setting up libptexenc1:amd64 (2019.20190605.51237-3build2) …
Setting up texlive-binaries (2019.20190605.51237-3build2) …
update-alternatives: using /usr/bin/xdvi-xaw to provide /usr/bin/xdvi.bin
(xdvi.bin) in auto mode
update-alternatives: using /usr/bin/bibtex.original to provide /usr/bin/bibtex
(bibtex) in auto mode
Setting up lmodern (2.004.5-6) …
Setting up texlive-base (2019.20200218-1) …
mktexlsr: Updating /var/lib/texmf/ls-R-TEXLIVEDIST…
mktexlsr: Updating /var/lib/texmf/ls-R-TEXMFMAIN…
mktexlsr: Updating /var/lib/texmf/ls-R…
mktexlsr: Done.
tl-paper: setting paper size for dvips to a4:
/var/lib/texmf/dvips/config/config-paper.ps
tl-paper: setting paper size for dvipdfmx to a4:
/var/lib/texmf/dvipdfmx/dvipdfmx-paper.cfg
tl-paper: setting paper size for xdvi to a4: /var/lib/texmf/xdvi/XDvi-paper
tl-paper: setting paper size for pdftex to a4:
/var/lib/texmf/tex/generic/config/pdftexconfig.tex
Setting up tex-gyre (20180621-3) …
Setting up texlive-plain-generic (2019.202000218-1) …
Setting up texlive-latex-base (2019.20200218-1) …
Setting up texlive-latex-recommended (2019.20200218-1) …
Setting up texlive-pictures (2019.20200218-1) …
Setting up texlive-fonts-recommended (2019.20200218-1) …
Setting up tipa (2:1.3-20) …
Regenerating '/var/lib/texmf/fmtutil.cnf-DEBIAN'… done.
Regenerating '/var/lib/texmf/fmtutil.cnf-TEXLIVEDIST'… done.
update-fmtutil has updated the following file(s):
        /var/lib/texmf/fmtutil.cnf-DEBIAN
        /var/lib/texmf/fmtutil.cnf-TEXLIVEDIST
If you want to activate the changes in the above file(s),
you should run fmtutil-sys or fmtutil.
Setting up texlive-latex-extra (2019.202000218-1) …
Setting up texlive-xetex (2019.20200218-1) …
Setting up rake (13.0.1-4) …
Setting up libruby2.7:amd64 (2.7.0-5ubuntu1.7) …
```

```
Setting up ruby2.7 (2.7.0-5ubuntu1.7) …
Setting up ruby (1:2.7+1) …
Processing triggers for fontconfig (2.13.1-2ubuntu3) …
Processing triggers for mime-support (3.64ubuntu1) …
Processing triggers for libc-bin (2.31-0ubuntu9.9) …
Processing triggers for man-db (2.9.1-1) …
Processing triggers for tex-common (6.13) …
Running updmap-sys. This may take some time… done.
Running mktexlsr /var/lib/texmf … done.
Building format(s) --all.
        This may take some time… done.
```

4. Convert to PDF (replace 000000000 with your student ID)

[ ]:
```
%env STUDENT_ID=260925886
!jupyter nbconvert --to pdf "/content/drive/MyDrive/Colab Notebooks/
↪260925886_Assignment_4.ipynb"
```

```
env: STUDENT_ID=260925886
```

5. Download the resulting PDF file. If you are using Chrome, you can do so by running the following code. On other browsers, you can download the PDF using the file mananger on the left of the screen (Navigate to the file > Right Click > Download).

[24]:
```
import os
from google.colab import files
files.download(f"/content/drive/MyDrive/Colab Notebooks/{os.
↪environ['STUDENT_ID']}_Assignment_4.pdf")
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-24-8fdbef91b2f5> in <module>
      1 import os
      2 from google.colab import files
----> 3 files.download(f"/content/drive/MyDrive/Colab Notebooks/{os.
↪environ['STUDENT_ID']}_Assignment_4.pdf")

/usr/local/lib/python3.9/dist-packages/google/colab/files.py in␣
↪download(filename)
    220   if not _os.path.exists(filename):
    221     msg = 'Cannot find file: {}'.format(filename)
--> 222     raise FileNotFoundError(msg)  # pylint: disable=undefined-variable
    223
    224   comm_manager = _IPython.get_ipython().kernel.comm_manager

FileNotFoundError: Cannot find file: /content/drive/MyDrive/Colab Notebooks/
↪260925886_Assignment_4.pdf
```

6. Verify that your PDF correctly displays your figures and responses.