**SALARY PREDICTOR**

**Project Report**

INDUSTRIAL TRAINING

Degree

**BACHELOR OF COMPUTER APPLICATION**

|  |  |
| --- | --- |
| PROJECT GUIDE:  **Mrs. Rohaila Naaz** | SUBMITTED BY:  **Ayush jain (TCA1901063)** |



**FACULTY OF ENGINEERING & COMPUTING SCIENCES**

**TEERTHANKER MAHAVEER UNIVERSITY, MORADABAD**

**Graphical user interface, text

Description automatically generated**

**ACKNOWLEDGEMENT**

I express my sincere indebtedness towards my guide Mrs Rohaila Naaz , Computer Science & Engineering, TMU Engineering College,Moradabad for his invaluable guidance, suggestions and supervision throughout the work. Without his kind patronage and guidance the project would not have taken shape. I would also like to express my gratitude and sincere regards for his kind approval of the project, time to time counselling and advices.

I owe sincere thanks to all the faculty members in the department of Computer Science & Engineering for their kind guidance and encouragement from time to time.

**Ayush jain**

**Place:**

**Date:**

**DECLARATION**

We hereby declare that this Project Report titled SALARY PREDICTION MODEL submitted by us and approved by our project guide,the College of Computing Sciences and Information Technology (CCSIT), Teerthanker Mahaveer University, Moradabad, is a bonafide work undertaken by us and it is not submitted to any other University or Institution for the award of any degree diploma / certificate or published any time before.

|  |  |  |
| --- | --- | --- |
|  |  | |
| **Student Name:** | Ayush jain | Signature |
|  |  |  |
|  |  |  |
|  |  |  |
| **Project Guide :** | Mrs Rohaila Naaz | Signature |

Table of Contents

[1 Project Title 6](#_Toc31652427)

[2 Problem Statement 6](#_Toc31652428)

[3 Project Description 6](#_Toc31652429)

[3.1 Scope of the Work 6](#_Toc31652430)

[3.2 Project Modules 6](#_Toc31652431)

[3.3 Context Diagram (High Level) 6](#_Toc31652432)

[4 Implementation Methodology 6](#_Toc31652433)

[5 Technologies to be used 6](#_Toc31652434)

[5.1 Software Platform 6](#_Toc31652435)

[5.2 Hardware Platform 7](#_Toc31652436)

[5.3 Tools, if any 7](#_Toc31652437)

[6 Advantages of this Project 7](#_Toc31652438)

[7 Assumptions, if any 7](#_Toc31652439)

[8 Future Scope and further enhancement of the Project 7](#_Toc31652440)

[9 Project Repository Location 7](#_Toc31652441)

[10 Definitions, Acronyms, and Abbreviations 8](#_Toc31652442)

[11 Conclusion 8](#_Toc31652443)

[12 References 9](#_Toc31652444)

**Appendix**

**A:Data Flow Diagram (DFD)**

**B: Screen Shots**

# SALARY PREDICTIOR

User can predict the salary by just giving required input. The model can’t predict output until all required input is filled. Filling wrong data or NAN data leads to error or wrong output.

# Problem Statement

Machine learning is the subset of artificial intelligence ,using machine learning we build algorithm or models which can predict future output and results based on past experience and trained dataset. By different output and training dataset our algorithm improves their learning with time. The project problem statement is to find or predict the salary for the employees. This model will help normal peoples to predict their salaries according to their experience .

# Project Description

The objective of this project to learn implementation of any machine learning model and understand the workflow of any model. This project is small contribution toward the community.

## Scope of the Work

User can predict the salary by just giving required input. The model can’t predict output until all required input is filled. Filling wrong data or NAN data leads to error or wrong output.

## Project Modules

1. **Data preparation and pre-processing** : In this module we fetch dataset and pre-process dataset by cleaning and removing the unrequired data.
2. **Data splitting :** In this module we split our dataset into 2 part ,one will train dataset which will use to train our model and second one is test dataset which will use score or test our model.
3. **Data visualization:** In this module we visualize dataset using matplotlib and seaborn.
4. **Modeling :**  In this module we will set our dataset in machine learning algorithm, predict output, score our model and teach model to improve on the biases of different output score.
5. **Model deployment : I**n this module we will deploy our model using Flask on web.

## Context Diagram (High Level)

Figure 1 summarizes the performance of our linear models for various choices of the regularization **1** parameter, while Table 2 lists the performance of the two best linear models on the test set. Note

that we were able to exceed the performance of the benchmark as well as Foxtrot’s model.

Chart, line chart

Description automatically generated

Figure 1:

Performance of the linear models on the withheld test set for a range of regularization parameters.

∙

|  |  |
| --- | --- |
| **Model** | **Error** |
| Unigram Lasso, *α* = 4 ∙ 10−6 | *£*6201 |
| Bigram Lasso, *α* = 4 10−6 | *£*6112 |

Table 2: Linear model performance

# Implementation Methodology

The implementation of this project is divided into 7 parts:

1. **Data preprocessing :** The first step is to import the dataset into our IDE, then filter the dataset by removing the unrequired data, duplicate row or data set and removing NaN data. Then filtered dataset is divided into subparts Train dataset and Test dataset.
2. **Choose the right estimator/algorithm for our problems:** After getting our data ready we choose preferred machine learning algorithm for problems and estimator.
3. **Fit the Model :** After choosing the right machine learning algorithm, we will fit our train dataset into the model and use it to make prediction on our data.
4. **Evaluating a Model**: After fitting data into model. We will be evaluating our model; it helps to find best machine learning model that represent our dataset and how well the chosen model work in future.
5. **Improve the model**: By putting different estimator we will score our model and try to improve the model on the biases of different output. At the end we will put all together.
6. **Deployment:**  This is the last step of our project where we deploy our model on web application using different framework but in this project, we used flask for deployment.

# Technologies to be used

**5.1** **Software Platform**

1. **Front-end**

Flask

1. **Back-end**

Mysql

1. **FrameWork**

Flask

**5.2** **Hardware Platform**

**RAM:**4GB

**Hard Disk:** 250Gb

**OS: windows, Linux, Mac etc**

**5.3** **Tools**

Python 3.2,

Anaconda,

Jupyter lab

Numpy ,

Pandas

**Machine Learning**

Linear Regression

Sklearn

Mean absolute error for finding error

Matplotlib

Seaborn

# Advantages of this Project

* User can predict the salary of an employe by just giving some required input expirence data according to their requierment
* User can access the model at time and from anywhere.
* There is not any prerequisite to use this project and the UI quite simple to use.
* It is free of cost

# Future Scope and further enhancement of the Project

The future of this project is, we can make a build app for this project and automate the project by automatically putting the input at different time to predict output. With the increase of different dataset, the accuracy of output will automatically increase.

# Project Repository Location

*<Guidelines: Mention the location of the latest Source Code and all related documents, like- Project Synopsis Report, Project Progress updates, Project Requirement Details, Project Report (Softcopy), Test Repository (all test scenarios, test cases etc.) used for Functional Testing of the project etc. The repository location must be somewhere in CCSIT-Lab>*

| **S#** | **Project Artifacts (softcopy)** | **Location** (Mention Lab-ID, Server ID, Folder Name etc.) | **Verified by Project Guide** | **Verified by Lab In-Charge** |
| --- | --- | --- | --- | --- |
|  | Project Synopsis Report (Final Version) |  | Name and Signature | Name and Signature |
|  | Project Progress updates |  | Name and Signature | Name and Signature |
|  | Project Requirement specifications |  | Name and Signature | Name and Signature |
|  | Project Report (Final Version) |  | Name and Signature | Name and Signature |
|  | Test Repository |  | Name and Signature | Name and Signature |
|  | Any other document, give details |  | Name and Signature | Name and Signature |

# Definitions, Acronyms, and Abbreviations

*<Guidelines: Provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the SRS. This information may be provided by reference to one or more appendices in the SRS or by reference to documents. This information may be provided by reference to an Annexure >*

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
|  |  |
|  |  |
|  |  |
|  |  |

# Conclusion

By building this project I learn machine learning and different tools for machine learning development. I also understand the workflow of machine learning algorithm and how to deploy any model. Also this project help me to enhance my resume and it will my small contribution towards community.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S#** | **Reference Details** | **Owner** | **Version** | **Date** |
|  | Project Synopsis | <Project Group ID> | 1.0 | DD-MM-YY |
|  | Project Requirements | <Project Group ID> |  |  |
|  |  |  |  |  |

**Annexure A**

**Data Flow Diagram (DFD)**

**(Mandatory)**

The entire working or the flow of the data can be divided into two groups for better understanding They are 1 DFD-L0, 2. DFD-L1

Send the data

Enter details

Salary Prediction

Server

User

**DFD-L0**

This is the initial idea for the data. The data has to be flown from user to server and form server to the user for the prediction of the Salary.

**Server**

Input details

Feed the values

**User**

Match values

With dataset

Predict the

salary

Send the details

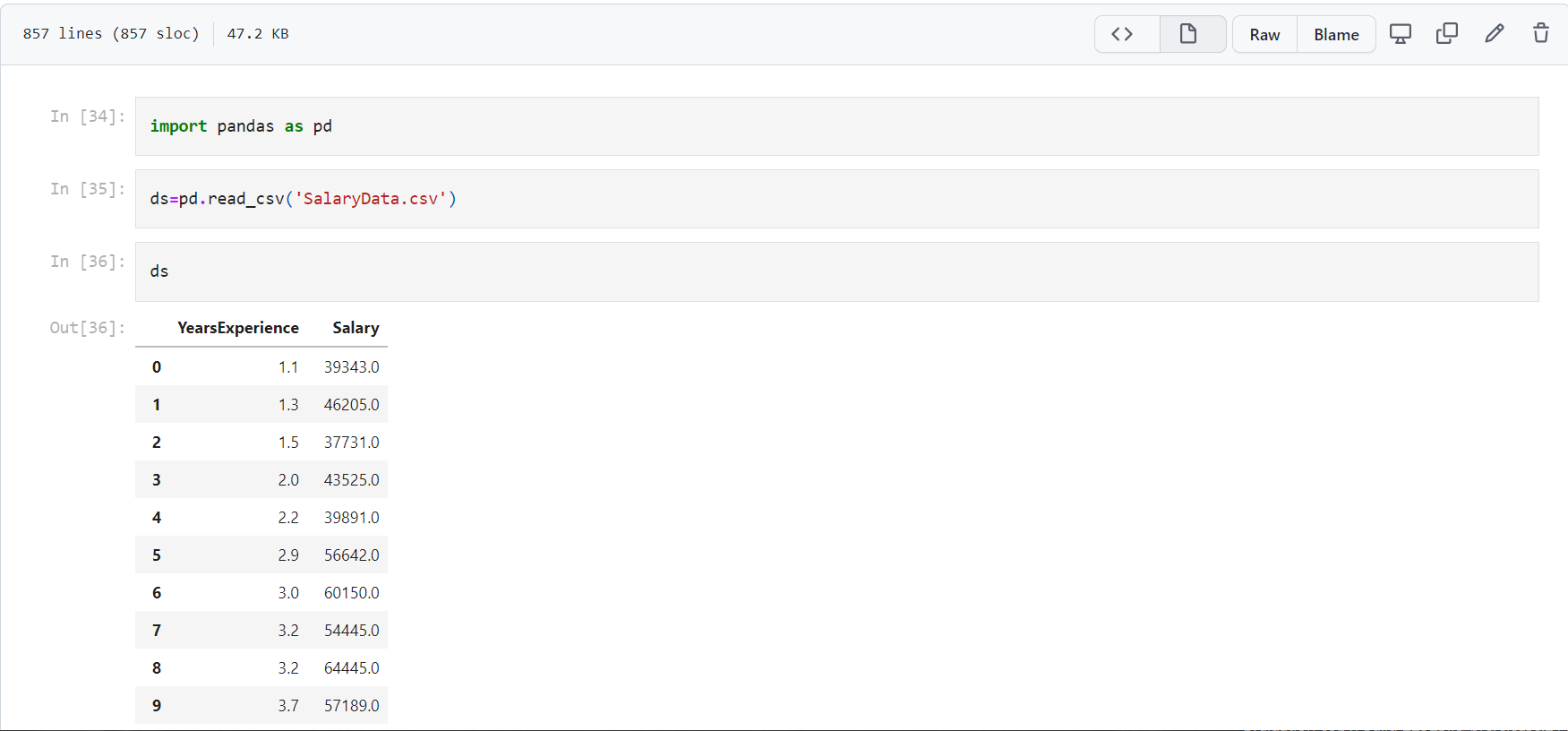
**DFD-L1**

****

This is the process or the idea where the data has been used to predict the salary by following several steps like registration (for new users), Feed the values(entering and storing values), Server(to store them), match the values(Finding probability) and finally predict the salary (Final result).

**Annexure B**

**Screen Shots**



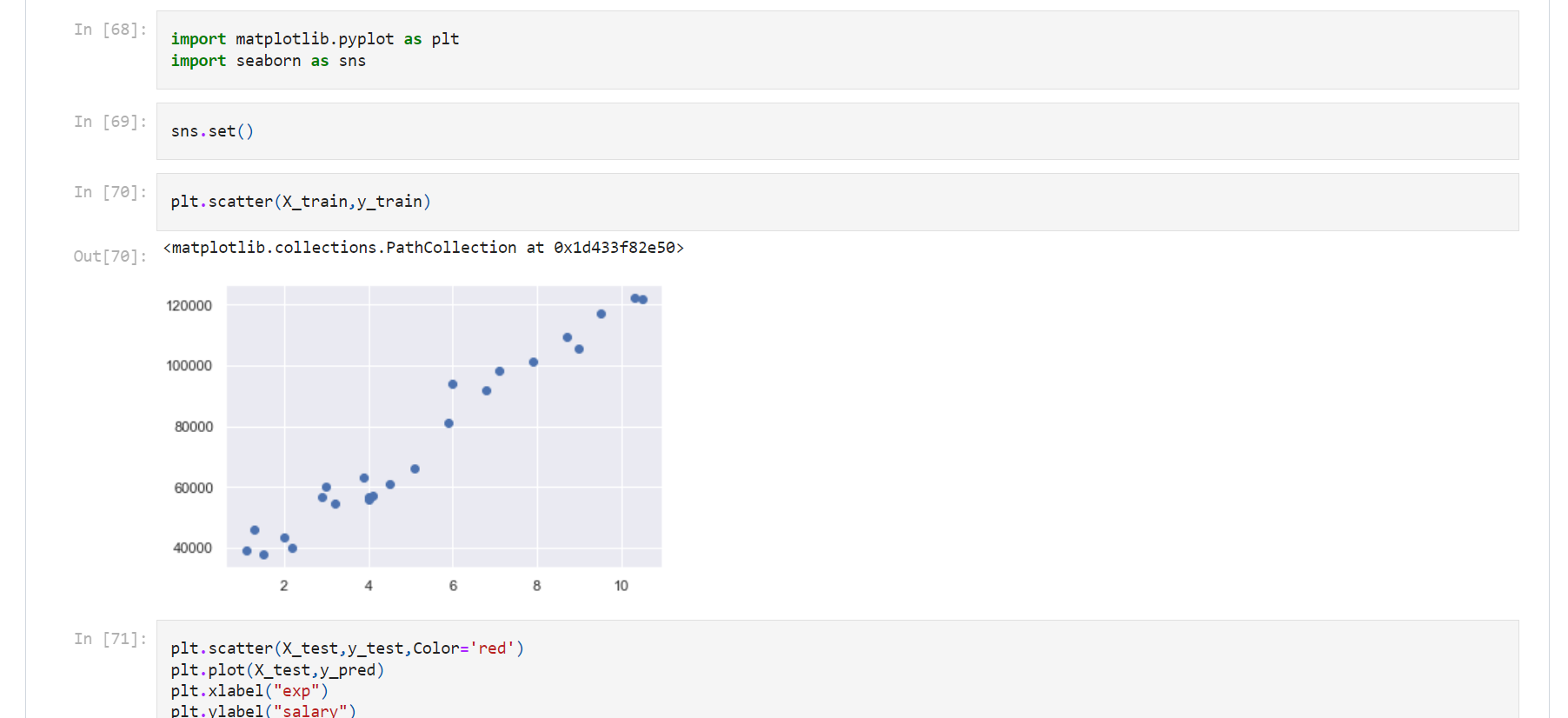


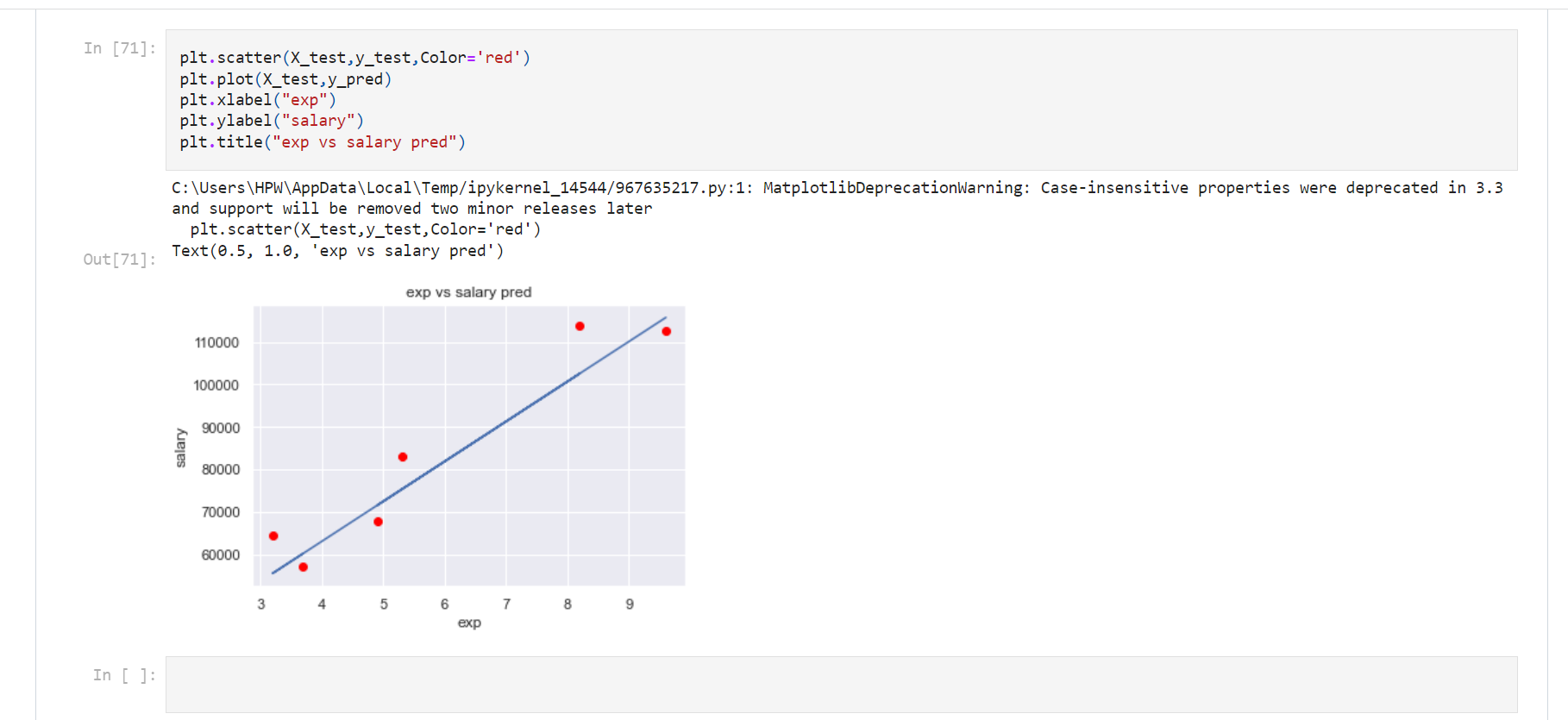


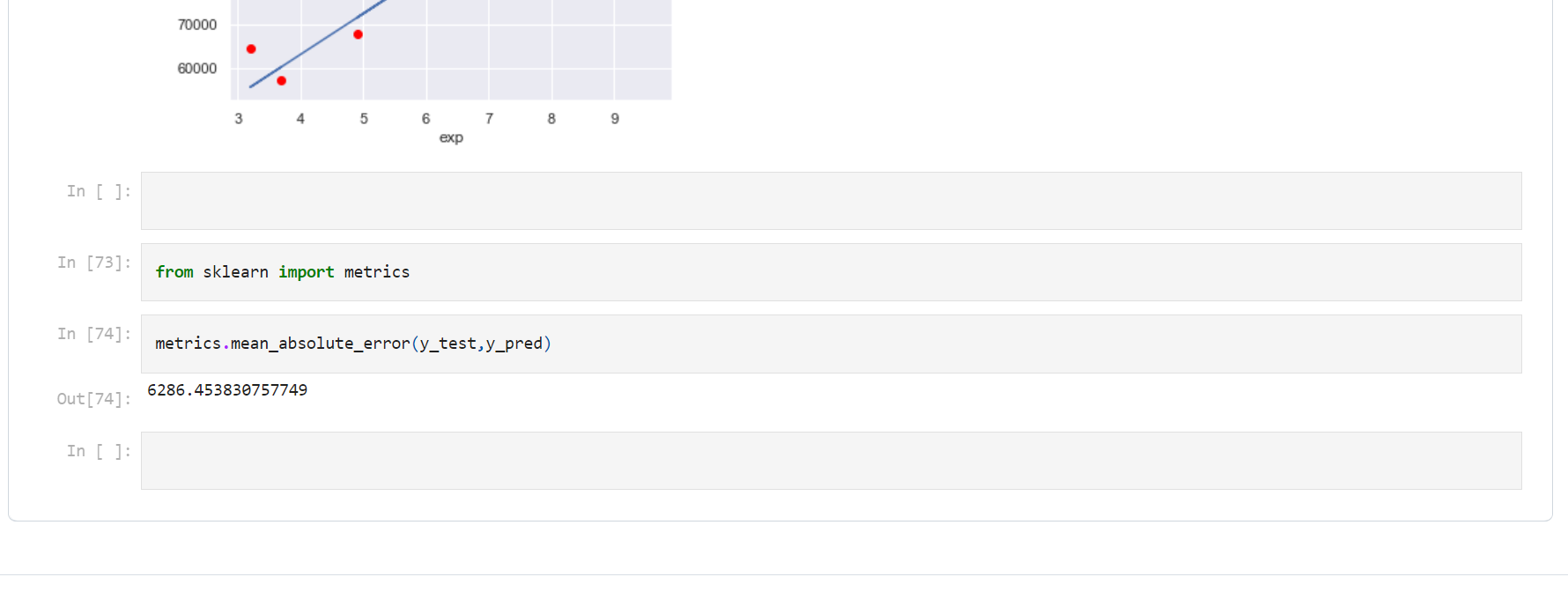


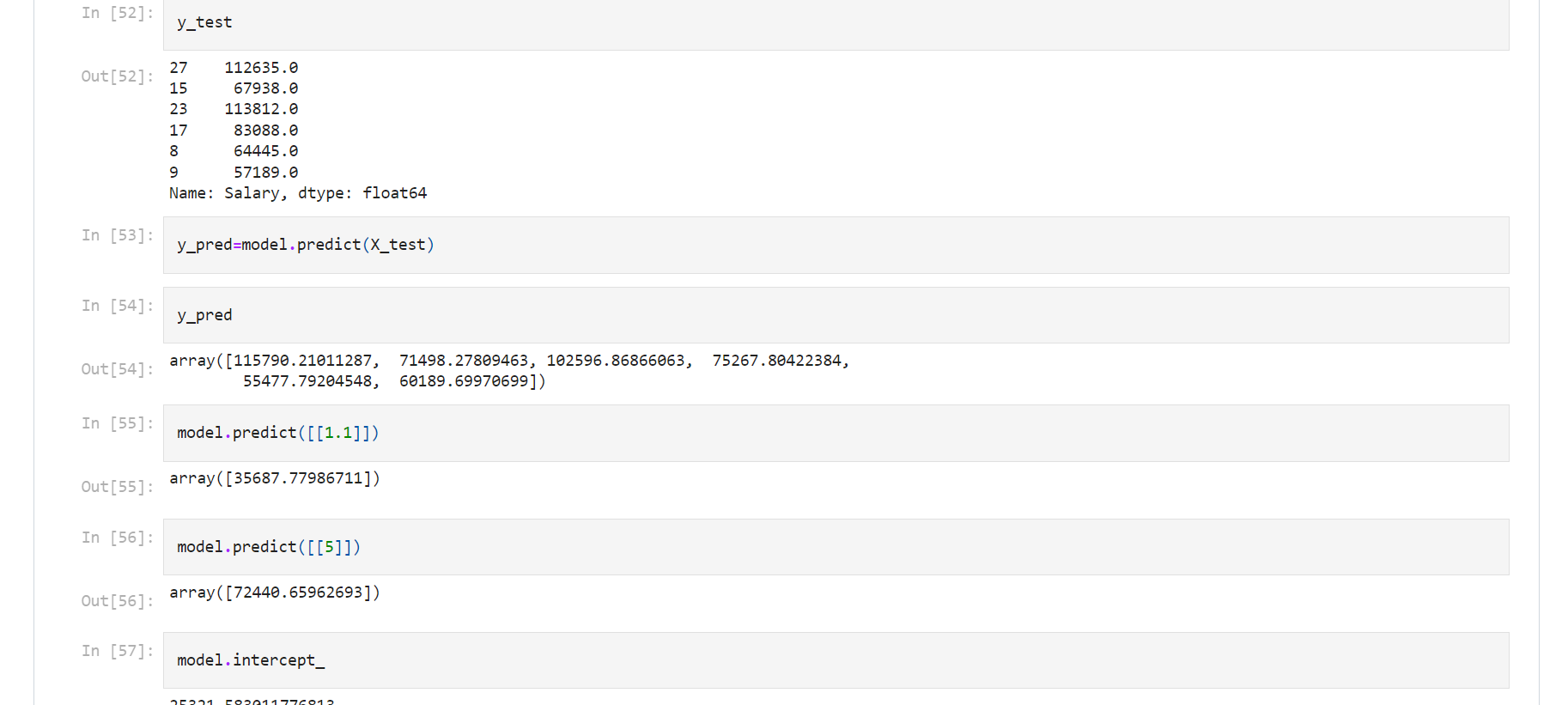
Graphical user interface, application

Description automatically generated









# Conclusions

Although we did not achieve the best results in the competition (the top performing method scored an average error of *£*3465 [1]), many of our methods scored in the top 10% of the competition with

our best method corresponding to the 6th percentile. These results are summarized in Table 7 which

lists the performance of the best model for each regression technique.

Most importantly, we demonstrated that linear models and information gain can be used to extract predictive features from text documents. Models based on these features consistently outperformed

the random forest benchmark provided by Adzuna. Additionally, we showed that while the inclusion of bigram features improved the linear models, it did not necessarily improve the predictive capabil**380** ities of our random forests. With our current feature selection mechanism, the bigram features were simply too sparse to impart any advantage over the unigram features.

Table 7: Performance of the best models for each regression technique. The “Aggregate RFR” model is a linear combination of the four best random forest models of section 6. This model was able to outperform all of our other models and achieved a ranking of 16th out of 294 competitors.

Due to time and computation constraints, the GBR models were not fully optimized. Additional work could concentrate on further improving these models, perhaps using gaussian process optimization to explore the high dimensional parameter space.

Additionally we would like to explore the possibility of further optimizing our feature extraction and selection techniques. In future work we would like to include features based on document statistics (document length, average number of words, Flesch-Kincaid readability statistics, etc) as well as features based on similarities between documents (perhaps using latent dirichlet allocation, an unsupervised learning technique which assumes that the distribution of words in a document is indicative of underlying categories [8]).

|  |  |
| --- | --- |
| **Model** | **Error** |
| Bigram Lasso, *α* = 4 ∙ 10−6 | *£*6112 |
| Unigram RFR, 800/200, *α* = 1 ∙ 10−4 | *£*5000 |
| Unigram GBR, 800/200, *λ* = 6*.*25 ∙ 10−2 | *£*5326 |
| Unigram Neural Network, *α* = 1*.*2 ∙ 10−7 | *£*5817 |
| Aggregate RFR | *£*4933 |

# References

**11.1. All Content used in this report is from**

1.https://trainings.internshala.com/progress/home/machine-learning

2.<https://www.kaggle.com/ronitf/salary-experience-uci>

3.<https://www.internshala.com>

4.<https://www.google.com/>

**11.2. All Pictures are from**

* <https://www.internshala.com/>
* <https://www.google.com/>
* <https://www.youtube.com/>

**11.3. Book I referred are**

* Hands-on Machine Learning with Scikit-learn & Tensorflow By Aurelien Geron
* Python Machine Learning by Sebastian Raschk

**11.4. GITHUB**

* https://github.com/Ayushjain2111/Salary\_Predictor