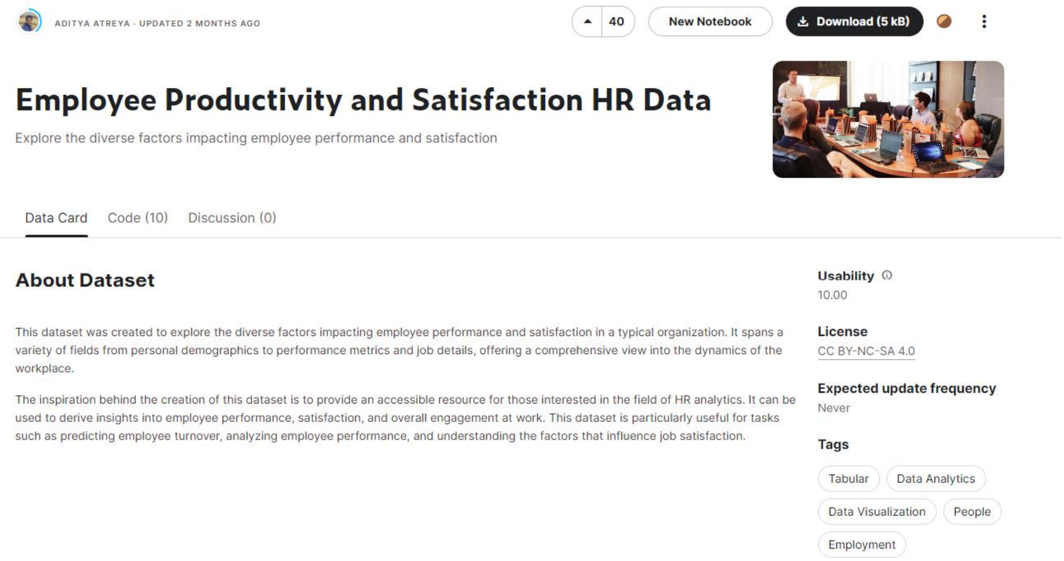
Ethan Feiza

LISUM25

10/04/2023

Data Glacier

1.) Select a data set to train a model that predicts employee salary.



ADITYA ATREYA · UPDATED 2 MONTHS AGO

Employee Productivity and Satisfaction HR Data

Explore the diverse factors impacting employee performance and satisfaction

About Dataset

This dataset was created to explore the diverse factors impacting employee performance and satisfaction in a typical organization. It spans a variety of fields from personal demographics to performance metrics and job details, offering a comprehensive view into the dynamics of the workplace.

The inspiration behind the creation of this dataset is to provide an accessible resource for those interested in the field of HR analytics. It can be used to derive insights into employee performance, satisfaction, and overall engagement at work. This dataset is particularly useful for tasks such as predicting employee turnover, analyzing employee performance, and understanding the factors that influence job satisfaction.

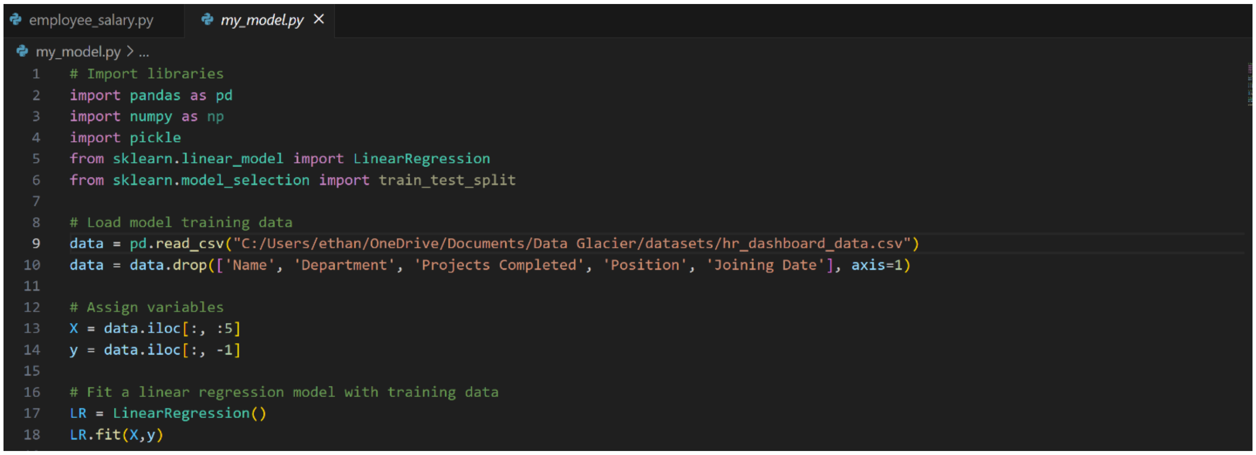
Usability
10.00

License
CC BY-NC-SA 4.0

Expected update frequency
Never

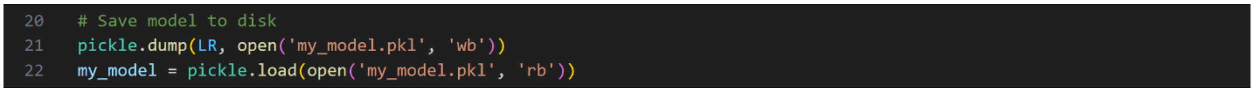
Tags
Tabular, Data Analytics, Data Visualization, People, Employment

2.) Download the data set and load it into Visual Studio. Import Python libraries. Fit a linear regression model on the data.



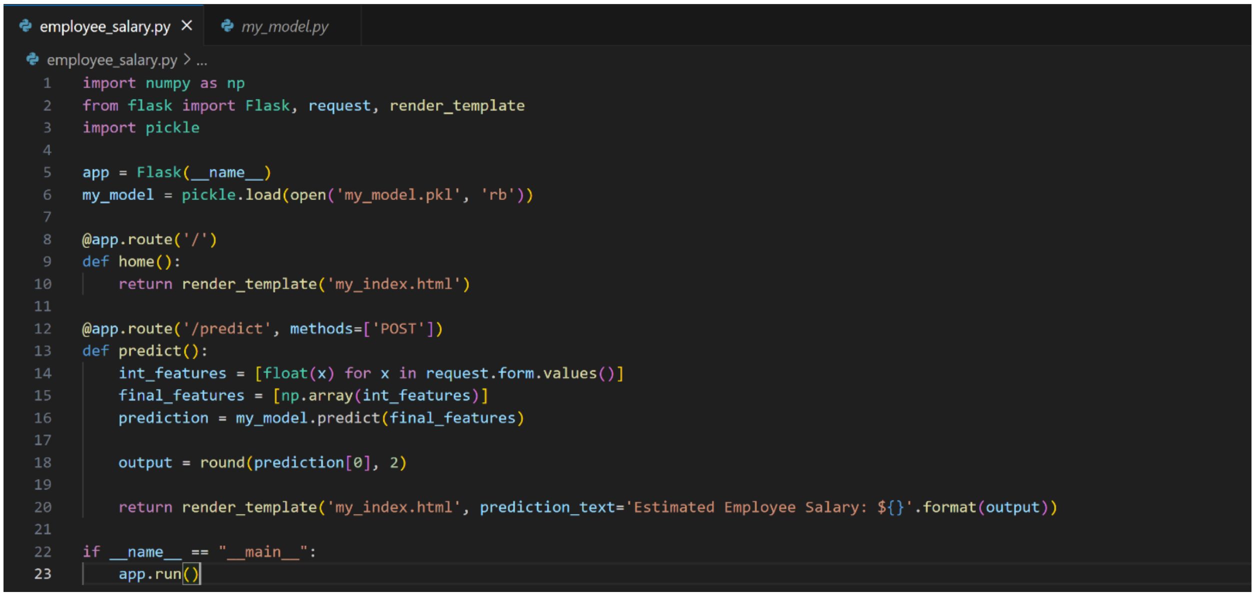
```python
# Import libraries
import pandas as pd
import numpy as np
import pickle
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split

# Load model training data
data = pd.read_csv("C:/Users/ethan/OneDrive/Documents/Data Glacier/datasets/hr_dashboard_data.csv")
data = data.drop(['Name', 'Department', 'Projects Completed', 'Position', 'Joining Date'], axis=1)

# Assign variables
X = data.iloc[:, :5]
y = data.iloc[:, -1]

# Fit a linear regression model with training data
LR = LinearRegression()
LR.fit(X,y)
```

3.) Save model to disk.



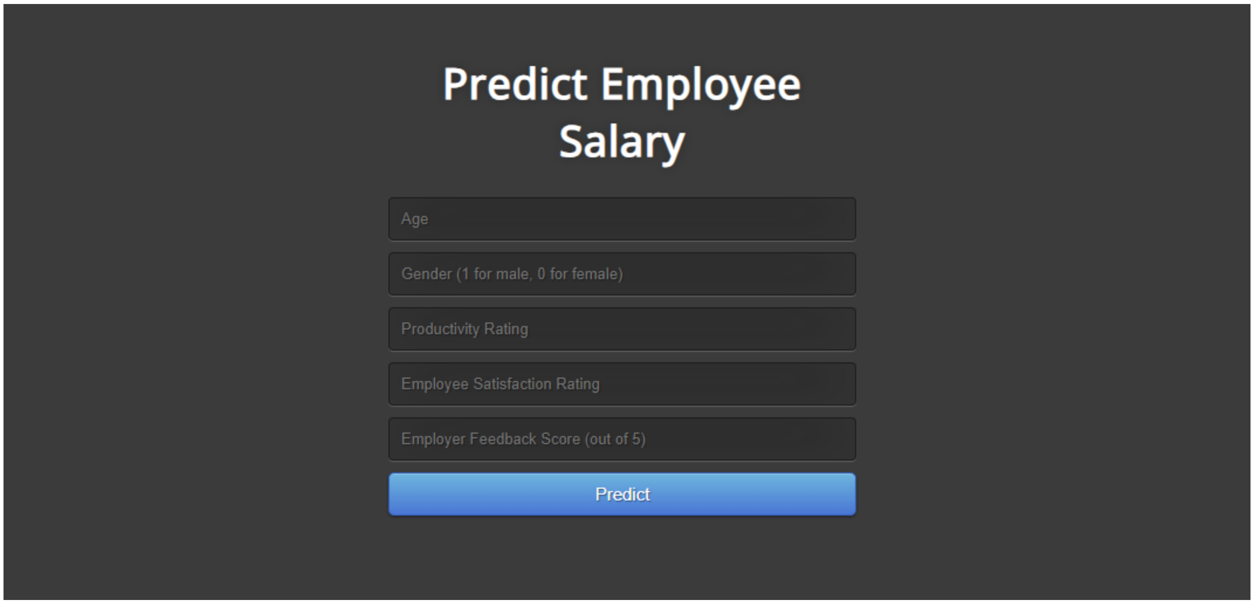
```python
# Save model to disk
pickle.dump(LR, open('my_model.pkl', 'wb'))
my_model = pickle.load(open('my_model.pkl', 'rb'))
```

4.) Deploy model with Flask.

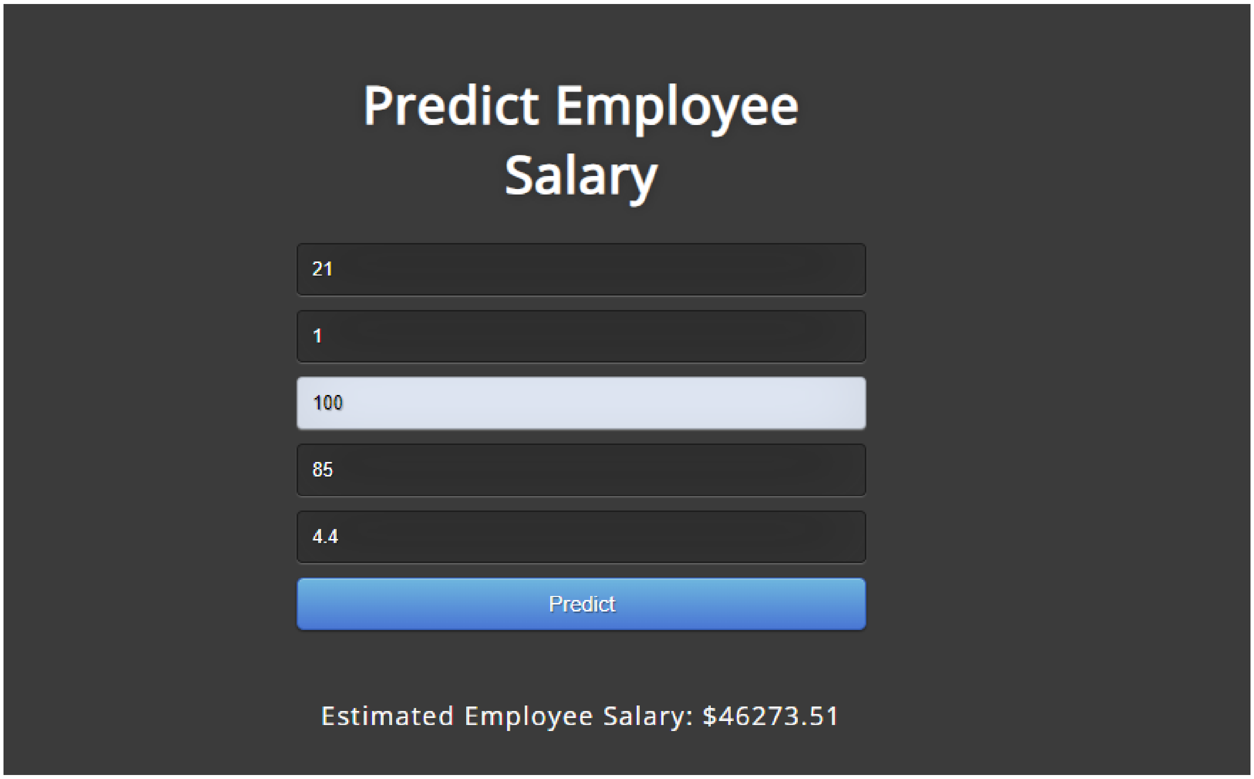


```python
import numpy as np
from flask import Flask, request, render_template
import pickle

app = Flask(__name__)
my_model = pickle.load(open('my_model.pkl', 'rb'))

@app.route('/')
def home():
    return render_template('my_index.html')

@app.route('/predict', methods=['POST'])
def predict():
    int_features = [float(x) for x in request.form.values()]
    final_features = [np.array(int_features)]
    prediction = my_model.predict(final_features)

    output = round(prediction[0], 2)

    return render_template('my_index.html', prediction_text='Estimated Employee Salary: ${}'.format(output))

if __name__ == "__main__":
    app.run()
```

5.) Locally run the Flask application.


Predict Employee Salary
Age
Gender (1 for male, 0 for female)
Productivity Rating
Employee Satisfaction Rating
Employer Feedback Score (out of 5)
Predict

6.) Test a prediction.



Predict Employee Salary

21

1

100

85

4.4

Estimated Employee Salary: $46273.51