

Week 5 Assignment

Name: **Enias Vontas**

Batch Code: **LISUM01**

Submission Date: **11th of July, 2021**

Submitted to:

<https://github.com/EniasVontas/Assignments/upload/main/Week5>

We will consider the same dataset, as the one used for Week 4, the 'insurance.csv' dataset, which was obtained from kaggle (<https://www.kaggle.com/mirichoi0218/insurance/>) and has 1338 rows and 7 features: age, gender, BMI, number of children, whether the person is a smoker or not, the region that they live and the insurance cost.

We will try to predict a person's insurance cost based on 5 of the above features, as the region that someone lives will not be considered.

The first five rows of our dataset and the types of features we have:

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

```
age          int64
sex          object
bmi          float64
children     int64
smoker       object
region       object
charges      float64
dtype: object
```

We observe that the 'bmi' features is a float, as well as the 'charges' feature. The 'sex', 'smoker' and 'region' features are objects. As we said, for simplicity, the feature 'region' will not be considered so we create dummy variables for the other two features. Our new dataset with the independent variables and our dependent ('charges') variable are:

	age	bmi	children	sex_male	smoker_yes
0	19	27.900	0	0	1
1	18	33.770	1	1	0
2	28	33.000	3	1	0
3	33	22.705	0	1	0
4	32	28.880	0	1	0

	charges
0	16884.92400
1	1725.55230
2	4449.46200
3	21984.47061
4	3866.85520

We would like to create our linear regression model and then use 'pickle' library to save it on our disk. So we create a .py file where we import the necessary libraries for our regression and the pickling that will follow. We load the data and create the data frame 'X' for our independent variables and the data frame 'Y' for our dependent variable. Then we modify the categorical features of our 'X' data frame to dummy variables. Lastly, we fit the linear regression model.

```
import pandas as pd
import numpy as np
import pickle
from sklearn.linear_model import LinearRegression

dataset = pd.read_csv("C:\\Program Files\\Git\\DataSets\\insurance.csv")

X = dataset[["age", "sex", "bmi", "children", "smoker"]]
Y = dataset[["charges"]]

X = pd.get_dummies(data=X, drop_first=True)

model = LinearRegression()
model.fit(X,Y)
```

We then pickle the model in order to save it to the disk and then load it again when we run the .py script in the command line.

```
pickle.dump(model, open('model.pkl', 'wb'))
model = pickle.load(open('model.pkl', 'rb'))
```

Next we create the app.py file in order to initialize the flask app and to render the template of the html file. We load our 'pickled' model and then create two routes, one for the template, and the other for the prediction, where we superimpose floating point numbers, as we can have BMIs with decimal points. Then we predict the insurance cost based on the parameters that we will enter on the Postman application.

```
import numpy as np
from flask import Flask, request, jsonify
import pickle
import pandas as pd

app = Flask(__name__)

@app.route('/', methods = ["GET", "POST"])
def home():
    if(request.method == "GET"):
        data = "hello world"
        return jsonify({'data':data})

@app.route('/predict')
def predict():
    model = pickle.load(open('model.pkl', 'rb'))
    age = request.args.get('age')
    bmi = request.args.get('bmi')
    children = request.args.get('children')
    sex = request.args.get('sex_male')
    smoker = request.args.get('smoker_yes')

    test_df = pd.DataFrame({'Age':[age], 'BMI':[bmi],
                           "No of Children":[children],
                           "Sex":[sex], "Smoker":[smoker]})
    pred_price = model.predict(test_df)

    return jsonify({'Insurance':str(pred_price)})

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

And the result from Postman is as shown below, for a male 30 year old individual, with 23.2 BMI, who has 1 child and is a non-smoker. The charges are 3504.21 \$.

Overview

GET http://127.0.0.1:50... ● + ...

http://127.0.0.1:5000/predict?age=30&bmi=23.2&children=1&sex_male=1&smoker_yes=0

GET ▼

http://127.0.0.1:5000/predict?age=30&bmi=23.2&children=1&sex_male=1&smoker_yes=0

Params ●

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	age	30
<input checked="" type="checkbox"/>	bmi	23.2
<input checked="" type="checkbox"/>	children	1
<input checked="" type="checkbox"/>	sex_male	1
<input checked="" type="checkbox"/>	smoker_yes	0
	Key	Value

Body

Cookies

Headers (4)

Test Results

Pretty

Raw

Preview

Visualize

JSON ▼

1

2

3

25

"Insurance": "[3504.20868753]"

25

Next, we would like to create an application through Heroku, where we can predict the amount of insurance someone will be charged. In order to do that, we create a Heorku account, and connect our GitHub repository (named 'heroku'), with Heorku.

EniasVontas / heroku

<> CodeIssuesPull requestsActionsProjectsWikiSecurityInsightsSettings

main1 branch0 tagsGo to fileAdd fileCode

EniasVontas Update README.md4cad0de now10 commits

static	Add files via upload	1 hour ago
templates	Add files via upload	1 hour ago
Procfile	Update Procfile	1 hour ago
README.md	Update README.md	now
app.py	Update app.py	27 minutes ago
insurance.csv	Add files via upload	1 hour ago
model.pkl	Add files via upload	1 hour ago
model.py	Add files via upload	1 hour ago
requirements.txt	Add files via upload	1 hour ago
runtime.txt	Update runtime.txt	1 hour ago

Personal > insurance-ch

GitHub EniasVontas/heroku

OverviewResourcesDeployMetricsActivityAccessSettings

Add this app to a pipeline

Create a new pipeline or choose an existing one and add this app to a stage in it.

Add this app to a stage in a pipeline to enable additional features

Pipelines let you connect multiple apps together and promote code between them.
[Learn more.](#)

Pipelines connected to GitHub can enable review apps, and create apps for new pull requests.
[Learn more.](#)

Choose a pipeline

Deployment method

Heroku Git
Use Heroku CLI

GitHub
Connected

Container Registry
Use Heroku CLI

App connected to GitHub

Code diffs, manual and auto deploys are available for this app.

Connected to EniasVontas/heroku by EniasVontas

Disconnect...

Releases in the [activity feed](#) link to GitHub to view commit diffs

And then we deploy the main branch of this repository:

Automatic deploys

Enables a chosen branch to be automatically deployed to this app.



You can now change your main deploy branch from "master" to "main" for both manual and automatic deploys, please follow the instructions [here](#).

Enable automatic deploys from GitHub

Every push to the branch you specify here will deploy a new version of this app. **Deploys happen automatically:** be sure that this branch is always in a deployable state and any tests have passed before you push. [Learn more](#).

Choose a branch to deploy

main

☐ Wait for CI to pass before deploy

Only enable this option if you have a Continuous Integration service configured on your repo.

Enable Automatic Deploys

Manual deploy

Deploy the current state of a branch to this app.

Deploy a GitHub branch

This will deploy the current state of the branch you specify below. [Learn more](#).

Choose a branch to deploy

main

Deploy Branch

The application has been created with url 'https://insurance-ch.herokuapp.com'. We would like to again compute the insurance charges for a 30 year old male individual, with 23.2 BMI, who has 1 child and is a non-smoker.

https://insurance-ch.herokuapp.com

Most Visited Getting Started

-*- coding: utf-8 -*- "" Created on Sat Jul 3 13:05:39 2021 @author: Enias ""

Predict Insurance Charges

Predict

I would fail to get a result through the Heroku app and unfortunately, I was not able to find the solution in time, before the end of the deadline.