



**Федеральное государственное бюджетное
образовательное учреждение
высшего образования
«Московский государственный
технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

Факультет «Информатика и вычислительная техника»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Технология машинного обучения»

Отчет по лабораторной работе №6
«Создание веб-приложения для демонстрации моделей машинного
обучения.»

Выполнил:
студент группы ИУ5-63Б

Коновалов М. В.

Подпись и дата:

Проверил:
преподаватель каф.
ИУ5

Гапанюк Ю.Е.

Подпись и дата:

Москва, 2023

Описание задания

Разработайте макет веб-приложения, предназначенного для анализа данных.

Вариант 1. Макет должен быть реализован для одной модели машинного обучения. Макет должен позволять:

- задавать гиперпараметры алгоритма,
- производить обучение,
- осуществлять просмотр результатов обучения, в том числе в виде графиков.

Вариант 2. Макет должен быть реализован для нескольких моделей машинного обучения. Макет должен позволять:

- выбирать модели для обучения,
- производить обучение,
- осуществлять просмотр результатов обучения, в том числе в виде графиков.

Для разработки рекомендуется использовать следующие (или аналогичные) фреймворки:

- [streamlit](#)
- [gradio](#)
- [dash](#)

Текст программы

```
import streamlit as st
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import mean_squared_error, r2_score, median_absolute_error
from sklearn import tree
import matplotlib.pyplot as plt
```

```
# загрузка данных
@st.cache_data
def load_data():
    data = pd.read_csv('Admission_Predict.csv')
    return data
```

```
# выбор модели для обучения
```

```

def select_model():
    models = ['Linear Regression', 'Decision Tree']
    model = st.selectbox('Select a model', models, key='my_selectbox')
    if model == 'Linear Regression':
        return LinearRegression()
    elif model == 'Decision Tree':
        return DecisionTreeRegressor(max_depth = 6, min_samples_leaf = 3,
min_samples_split = 2)

# обучение модели
def train_model(model, X_train, y_train):
    model.fit(X_train, y_train)
    return model

# просмотр результатов обучения
def show_results(model, X_test, y_test):
    y_pred = model.predict(X_test)
    st.write('Mean Squared Error:', mean_squared_error(y_test, y_pred))
    st.write('R2 Score:', r2_score(y_test, y_pred))
    st.write('Median absolute error:', median_absolute_error(y_test, y_pred))

def show_plot(model, X_test):
    y_pred = model.predict(X_test)
    if isinstance(model, DecisionTreeRegressor):
        fig, ax = plt.subplots()
        tree.plot_tree(model, ax=ax)
        st.pyplot(fig)
    elif isinstance(model, LinearRegression):
        plt.plot(X_test, y_pred, color='red')
        plt.xlabel('X')
        plt.ylabel('Y')
        plt.title('Regression')
        st.pyplot()

def main():
    st.title('Machine Learning Model Training')
    data = load_data()
    st.write(data.head())
    X = data.drop(['Chance of Admit '], axis=1)
    y = data['Chance of Admit ']
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, shuffle=False,
random_state=45)

    model = select_model()
    trained_model = train_model(model, X_train, y_train)
    show_results(trained_model, X_test, y_test)
    show_plot(trained_model, X_test)

```

```
if __name__ == '__main__':  
    main()
```

Экранные формы

localhost

tab6 · S...

Machine Learning Model Training

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of /
0	1	337	118	4	4.5	4.5	9.65	1	
1	2	324	107	4	4	4.5	8.87	1	
2	3	316	104	3	3	3.5	8	1	
3	4	322	110	3	3.5	2.5	8.67	1	
4	5	314	103	2	2	3	8.21	0	

Select a model

Linear Regression

Mean Squared Error: 0.004804385696652735

R2 Score: 0.7832591660093955

Median absolute error: 0.03884852585660076

PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()
...     plt.plot(x, y, 'r', label='y=x')
```

Machine Learning Model Training

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of /
0	1	337	118	4	4.5	4.5	9.65	1	
1	2	324	107	4	4	4.5	8.87	1	
2	3	316	104	3	3	3.5	8	1	
3	4	322	110	3	3.5	2.5	8.67	1	
4	5	314	103	2	2	3	8.21	0	

Select a model

Linear Regression

Mean Squared Error: 0.004804385696652735

R2 Score: 0.7832591660093955

Median absolute error: 0.03884852585660076

PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()
...     plt.plot(x, y, 'r', label='y=x')
```

localhost

tab6 · S...

Machine Learning Model Training

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of /
0	1	337	118	4	4.5	4.5	9.65	1	
1	2	324	107	4	4	4.5	8.87	1	
2	3	316	104	3	3	3.5	8	1	
3	4	322	110	3	3.5	2.5	8.67	1	
4	5	314	103	2	2	3	8.21	0	

Select a model

Linear Regression

Mean Squared Error: 0.004804385696652735

R2 Score: 0.7832591660093955

Median absolute error: 0.03884852585660076

PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()
... 
```

Machine Learning Model Training

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of /
0	1	337	118	4	4.5	4.5	9.65	1	
1	2	324	107	4	4	4.5	8.87	1	
2	3	316	104	3	3	3.5	8	1	
3	4	322	110	3	3.5	2.5	8.67	1	
4	5	314	103	2	2	3	8.21	0	

Select a model

Linear Regression

Mean Squared Error: 0.004804385696652735

R2 Score: 0.7832591660093955

Median absolute error: 0.03884852585660076

PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()
... 
```