auto-water-supply-lr

June 28, 2024

[ ]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
```

[ ]:
```
data = pd.read_csv('/content/drive/MyDrive/PUMPING/data.csv')
```

[ ]:
```
x= ['moisture','temp']
y = ['pump']
```

[ ]:
```
LR = LinearRegression()
```

[ ]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

data = pd.read_csv('/content/drive/MyDrive/PUMPING/data.csv')

# Extract features and target variable as dataframes, not lists
x = data[['moisture','temp']]  # Use double brackets to select multiple columns
y = data[['pump']]

LR = LinearRegression()

LR.fit(x,y)
```

[ ]:
```
LinearRegression()
```

[ ]:
```
LR.predict([[500,20]])
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does
not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```

[ ]:
```
array([[0.56686135]])
```

[ ]:
```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Load your data (replace with your actual file path)
data = pd.read_csv('/content/drive/MyDrive/PUMPING/data.csv')

# Extract features and target variable
x = data[['moisture','temp']]
y = data[['pump']]

# Create and fit the linear regression model
LR = LinearRegression()
LR.fit(x, y)

# Get user input for moisture and temperature
user_moisture = float(input("Enter moisture value: "))
user_temp = float(input("Enter temperature value: "))

# Create a DataFrame for user input
user_input = pd.DataFrame({'moisture': [user_moisture], 'temp': [user_temp]})

# Make the prediction
prediction = LR.predict(user_input)

print("Predicted pump value:", prediction[0][0])

if prediction[0][0] < 0.5:
    print("Pump is off")
else:
    print("Pump is on")
```

```
Enter moisture value: 758
Enter temperature value: 25
Predicted pump value: 0.9140422424788589
Pump is on
```