

Лабораторна робота №8

Ресурси Keras. TensorFlow. Навчання лінійної регресії

Мета: дослідження ресурсу Keras і TensorFlow. Застосування TensorFlow

Хід роботи

Завдання. Використовуючи засоби TensorFlow, реалізувати код наведений нижче та дослідити структуру розрахункового алгоритму.

Лістинг коду:

```
import numpy as np
import tensorflow as tf
import matplotlib.pyplot as plt

# Встановлюємо параметри
n_samples = 1000 # кількість точок
batch_size = 100 # розмір міні-батчу
num_steps = 20001 # кількість ітерацій

# Генеруємо дані
X_data = np.random.uniform(0, 1, (n_samples, 1)) # вхідні дані x
y_data = 2 * X_data + 1 + np.random.normal(0, 0.2, (n_samples, 1)) # цільові дані y

# Створюємо placeholders для вхідних даних та цілей
X = tf.placeholder(tf.float32, shape=(batch_size, 1), name='X')
y = tf.placeholder(tf.float32, shape=(batch_size, 1), name='y')

# Оголошуємо змінні моделі
with tf.variable_scope('linear-regression'):
    k = tf.get_variable('k', shape=(1, 1),
        initializer=tf.random_normal_initializer())
    b = tf.get_variable('b', shape=(1, 1),
        initializer=tf.random_normal_initializer())

# Описуємо модель
y_pred = tf.matmul(X, k) + b

# Визначаємо функцію втрат
loss = tf.reduce_sum((y_pred - y) ** 2)

# Вибираємо оптимізатор
optimizer = tf.train.GradientDescentOptimizer(0.01).minimize(loss)

# Навчання моделі
display_step = 100 # періодичність виведення результатів

with tf.Session() as sess:
    # Ініціалізуємо змінні
    sess.run(tf.global_variables_initializer())
```

					ДУ «Житомирська політехніка».24.121.16.000 – Лр8			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Некритий В.Ю.			Звіт з лабораторної роботи	Лім.	Арк.	Аркушів
Перевір.		Іванов Д.А.					1	2
Керівник						ФІКТ Гр. ІПЗ-21-5		
Н. контр.								
Зав. каф.								

```

for i in range(num_steps):
    # Вибірка випадкових індексів для батчу
    indices = np.random.choice(n_samples, batch_size)
    X_batch, y_batch = X_data[indices], y_data[indices]

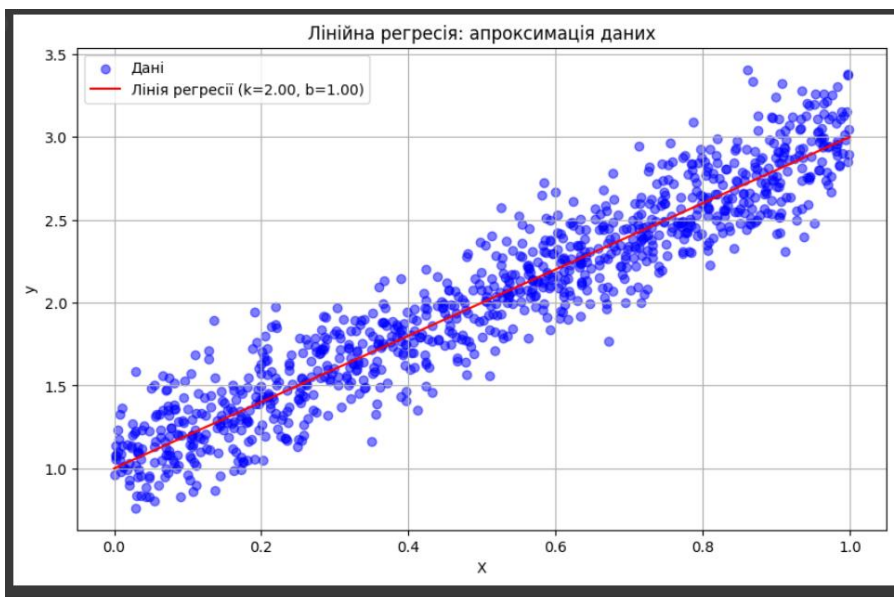
    # Виконуємо оптимізацію
    _, loss_val, k_val, b_val = sess.run([optimizer, loss, k, b],
                                         feed_dict={X: X_batch, y: y_batch})

    # Виводимо проміжні результати
    if (i + 1) % display_step == 0:
        print(f"Епоха {i + 1}: втрата={loss_val:.8f}, k={k_val[0][0]:.4f},
b={b_val[0]:.4f}")

# Візуалізація
plt.figure(figsize=(10, 6))
plt.scatter(X_data, y_data, label="Дані", alpha=0.5, color='blue')
x_plot = np.linspace(0, 1, 100).reshape(-1, 1)
y_plot = k_val * x_plot + b_val
plt.plot(x_plot, y_plot, color='red', label=f'Лінія регресії (k={k_val[0][0]:.2f},
b={b_val[0]:.2f})')
plt.xlabel("x")
plt.ylabel("y")
plt.title("Лінійна регресія: апроксимація даних")
plt.legend()
plt.grid()
plt.show()

```

Результат виконання:



Висновок: реалізував код використовуючи засоби TensorFlow.

		Некритий В.Ю.			ДУ «Житомирська політехніка».24.121.16.000 – Лр8	Арк.
		Іванов Д.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		2