

Федеральное агентство связи

Федеральное государственное бюджетное образовательное  
учреждение высшего образования «Сибирский государственный  
университет  
телекоммуникаций и  
информатики» (СибГУТИ)

## **Лабораторная работа №4**

Выполнил: студент IV курса ИВТ,

гр. ИП-813

Бурдуковский И.А.

Проверила: ассистент кафедры

ПМиК

Морозова К.И.

Новосибирск, 2021 г.

## Оглавление

Задание.....	3
Выполнение .....	4
Код программы.....	5

## Задание

Целью данной лабораторной работы является разработка нейронной сети для решения задачи классификации или регрессии в зависимости от набора данных в рамках варианта. Лабораторная работа предполагает разработку на языке программирования Python с использованием библиотеки Keras.

Определение цены недвижимости (Boston Housing price regression dataset) Все наборы данных доступны по ссылке: <https://keras.io/api/datasets/>

При разработке нейронной сети следует соблюсти наличие необходимых составляющих исходя из следующих вариантов:

Нейросеть должна состоять из четырех полносвязных слоёв, обязательное использование GaussianDropout, в качестве оптимизатора использовать SGD.

Выбор количества нейронов на всех внутренних слоях, функций активации и других параметров должен быть обусловлен оптимальностью работы модели.

# Выполнение

Результат работы нейросети:

```
Epoch 999/1000
11/11 [=====] - 0s 3ms/step - loss: 9.9652 - mae: 2.3698 - accuracy: 0.0000e+00 - val_loss: 9.0611 - val_mae: 2.1289 - val_accuracy: 0.0000e+00
Epoch 1000/1000
11/11 [=====] - 0s 3ms/step - loss: 8.4937 - mae: 2.1911 - accuracy: 0.0000e+00 - val_loss: 9.4032 - val_mae: 2.2674 - val_accuracy: 0.0000e+00
4/4 [=====] - 0s 674us/step - loss: 10.7441 - mae: 2.1796 - accuracy: 0.0000e+00
Предсказанная цена и фактическая
predict    Заданные значения
25387.56$ ~ 26400.00$
31452.50$ ~ 37000.00$
28104.56$ ~ 24100.00$
26487.19$ ~ 24500.00$
25366.04$ ~ 23700.00$
10140.89$ ~ 7000.00$
24005.25$ ~ 22200.00$
25088.75$ ~ 23300.00$
15440.83$ ~ 15600.00$
12678.77$ ~ 13400.00$
```

## Код программы

```
import random
from tensorflow import metrics
from tensorflow.keras.datasets import cifar10
from sklearn.preprocessing import StandardScaler
from tensorflow.keras import models, layers, constraints
from tensorflow.keras import utils
from keras.constraints import maxnorm
from tensorflow.keras.datasets import boston_housing
import scipy
from sklearn.preprocessing import StandardScaler
from tensorflow.keras import models, layers

(X_train, y_train), (X_test, y_test) = boston_housing.load_data(seed=1)
# print(X_train[0], y_train[0])
scaler = StandardScaler()
scaler.fit(X_train)
X_train_scaled = scaler.transform(X_train)
X_test_scaled = scaler.transform(X_test)
# print(X_train_scaled[0])

model = models.Sequential()
model.add(layers.Dense(64, activation='relu', input_shape=[X_train.shape[1]]))
model.add(layers.GaussianDropout(0.2))
model.add(layers.Dense(64, activation='relu'))
model.add(layers.GaussianDropout(0.2))
model.add(layers.Dense(1))

model.compile(optimizer='adam', loss='mse', metrics=['mae', 'accuracy'])
# model.compile(optimizer='sgd', loss='mse', metrics=['mae'])
model.fit(X_train_scaled, y_train, validation_split=0.2, epochs=500, batch_size=32)
model.evaluate(X_test_scaled, y_test)

num = 10;
to_predict = X_test_scaled[:num]
predictions = model.predict(to_predict)

print("Предсказанная цена и фактическая")
print("predict    Заданные значения");
for i in range(num):
    print('%.2f$ ~ %.2f$' %(predictions[i]*1000, y_test[i]*1000))

# print(X_test_scaled.shape)
# to_predict_sum = X_test_scaled
# predictions_sum = model.predict(to_predict_sum)
# print("Проц = ", metrics.mean_absolute_error(y_test, predictions_sum))
))
```