Nama Dosen : Teguh Iman Hermanto, M.Kom

Mata Kuliah : Machine Learning 2

Pembahasan : Implementasi ANN KOTLIN

Pokok Pemb : Implementasi ANN dengan Tensorflow Lite dan Kotlin

Import Library dan membuat Array

```
1 import tensorflow as tf
2 import numpy as np
3 from tensorflow import keras
4 import matplotlib.pyplot as plt
```

```
1 jarak = np.array([2.0, 4.0, 6.0, 8.0, 10.0, 12.0, 14.0], dtype=float)
2 bbm = np.array([6.0, 10.0, 14.0, 18.0, 22.0, 26.0, 30.0], dtype=float)
```

```
plt.plot(jarak, bbm, 'o-')
plt.xlabel('jarak')
plt.ylabel('bbm')
plt.title('Plot jarak vs bbm')
plt.grid(True)
plt.show()
```

Membuat model dengan ANN

```
1 model = tf.keras.Sequential([
2 keras.layers.Dense(units=1, input_shape=[1])])
```

```
1 model.compile(
2    optimizer='sgd',
3    loss='mean_squared_error')
```

Menjalankan model dan simulasi data baru

```
1 model.fit(jarak, bbm, epochs=150)
```

```
print(model.predict(np.array([13.0])))
```

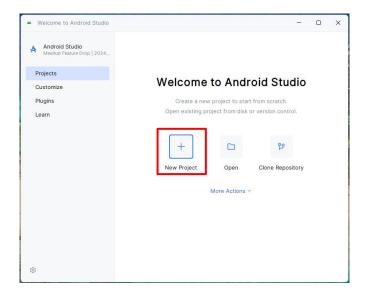
```
1 # membuat prediksi dengan plot
2 predicted_bbm = model.predict(jarak)
4 # Plot Dataset
5 plt.figure(figsize=(10, 6))
6 plt.plot(jarak, bbm, 'o-', label='Dataset')
8 # Plot hasil prediksi
9 plt.plot(jarak, predicted_bbm, 'x--', label='Prediksi')
10
11 # Plot nilai baru
12 new_jarak = np.array([13.0])
13 predicted_new_bbm = model.predict(new_jarak)
14 plt.plot(new_jarak, predicted_new_bbm, 'ro', markersize=8, label='Hasil Prediksi')
15
16
17 plt.xlabel('jarak')
18 plt.ylabel('bbm')
19 plt.title('Plot jarak vs bbm dengan Prediksi')
20 plt.grid(True)
21 plt.legend()
22 plt.show()
```

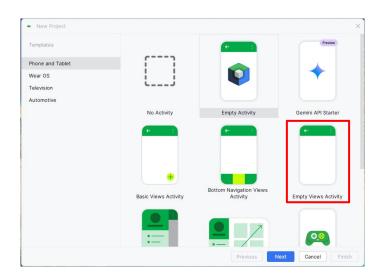
Menyimpan Model

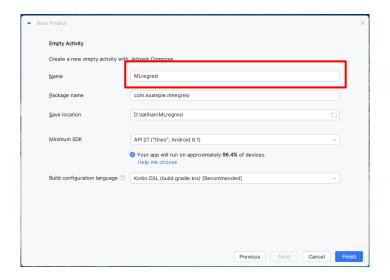
```
1 keras_file = "linear.h5"
2 tf.keras.models.save_model(model, keras_file)
```

```
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()
open("linear.tflite", "wb").write(tflite_model)
```

Membuat project baru di Android Studio







```
ML MLregresi Version control
                                                                                                                                                                              C
                                                                MainActivity.kt
     ∨ Comunication ✓ Maragresi [:] D:\latinan\MLregresi
                                                                                                                                                                              £2)
        > 🗀 .gradle
                                                                      package com.example.mlregresi
                                                                                                                                                                              > 🗀 .idea
        v 🗀 app
                                                                                                                                                                              > androidTest
                                                                        class MainActivity : ComponentActivity() {
                                                                           override fun onCreate(savedInstanceState: Bundle?) {
             ∨ 🦳 main
                                                                                super.onCreate(savedInstanceState)
               java
                                                                                enableEdgeToEdge()
                  ∨ 🗀 com
                                                                                setContent {
                                                                                    MLregresiTheme {
                                                                                        Scaffold(modifier = Modifier.fillMaxSize()) { innerPadding ->
                                                                                            Greeting(
name = "Android",
                          MainActivity.kt
                                                                                                 modifier = Modifier.padding(innerPadding)
                 M AndroidManifest.xml
             > 🗀 test
            aitianore .
                                                                                   }
            abuild.gradle.kts
             ≡ proguard-rules.pro
        > 🗀 gradle
          .qitiqnore
          € build.gradle.kts
                                                                        fun Greeting(name: String, modifier: Modifier = Modifier) {
           gradle.properties

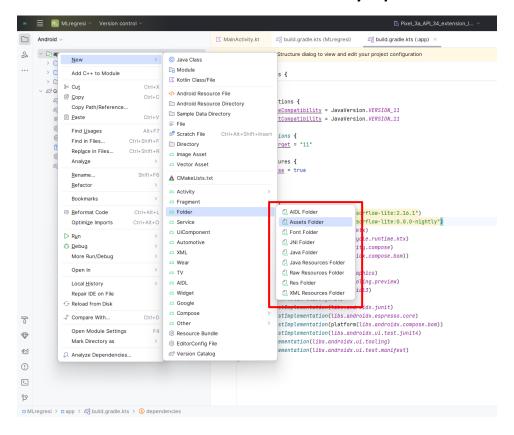
    gradlew

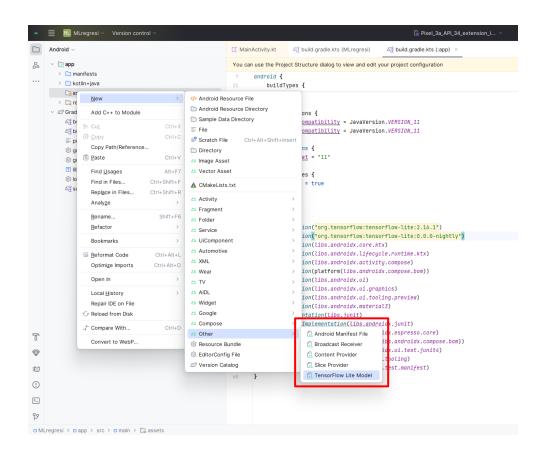
                                                                                text = "Hello $name!".
           ≡ gradlew.bat
                                                                                modifier = modifier
          (i) local properties
          a settings.gradle.kts
                                                                       }
       \Pi
        Scratches and Consoles
                                                                        @Preview(showBackground = true)
                                                                        fun GreetingPreview() {
∌
                                                                           MLregresiTheme {
(!)
                                                                                Greeting("Android")
>_
ဗှ
□ MLregresi > app > src > main > java > com > example > mlregresi > ☐ MainActivit
                                                                                Importing 'MLregresi' Gradle Project 🕳
                                                                                                                                 Show all (3) 1:1 LF UTF-8
```

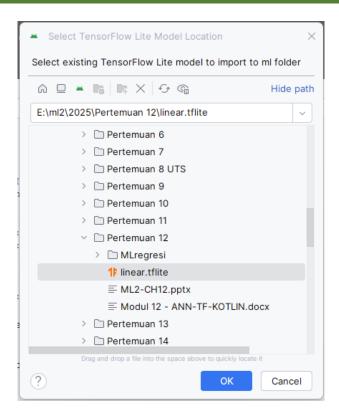
Menambahkan tensorflow package pada build.gradle

```
MainActivity.kt
                   € build.gradle.kts (MLregresi)
                                                  € build.gradle.kts (:app) ×
Gradle files have changed since last project sync. A project sync may be necessary for the IDE to work properly.
                                                                                               Sync Now
                                                                                                          Ignore these changes
       dependencies {
           implementation("org.tensorflow:tensorflow-lite:2.16.1")
           implementation("org.tensorflow:tensorflow-lite:0.0.0-nightly")
44
           implementation(libs.androidx.core.ktx)
           implementation(libs.androidx.lifecycle.runtime.ktx)
           implementation(libs.androidx.activity.compose)
           implementation(platform(libs.androidx.compose.bom))
           implementation(libs.androidx.ui)
           implementation(libs.androidx.ui.graphics)
           implementation(libs.androidx.ui.tooling.preview)
           implementation(libs.androidx.material3)
           testImplementation(libs.junit)
           androidTestImplementation(libs.androidx.junit)
           androidTestImplementation(libs.androidx.espresso.core)
           androidTestImplementation(platform(libs.androidx.compose.bom))
           androidTestImplementation(libs.androidx.ui.test.junit4)
           debugImplementation(libs.androidx.vi.tooling)
           debugImplementation(libs.androidx.vi.test.manifest)
      }
60
```

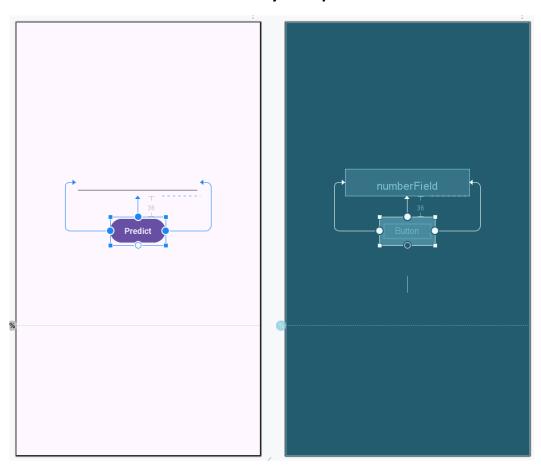
Menambahkan folder assets untuk menyimpan model







Membuat Layout Aplikasi



```
activity_main.xml
 <?xml version="1.0" encoding="utf-8"?>
 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"</p>
     xmlns:app="http://schemas.android.com/apk/res-auto"
     xmlns:tools="http://schemas.android.com/tools"
     android:layout_width="match_parent"
     android:layout_height="match_parent"
     tools:context=".MainActivity">
     <androidx.constraintlayout.widget.Guideline</pre>
         android:id="@+id/guideline"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:orientation="horizontal"
         app:layout_constraintGuide_percent="0.7" />
     <EditText
         android:id="@+id/numberField"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_marginBottom="220dp"
         android:ems="10"
         android:inputType="number"
         app:layout_constraintBottom_toTopOf="@id/guideline"
         app:layout_constraintEnd_toEndOf="parent"
         app:layout_constraintHorizontal_bias="0.497"
         app:layout_constraintStart_toStartOf="parent"
         tools:ignore="MissingConstraints" />
     <Button
         android:id="@+id/btnPredict"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_marginTop="36dp"
         android:text="Predict"
         app:layout_constraintEnd_toEndOf="@+id/numberField"
         app:layout_constraintStart_toStartOf="@+id/numberField"
         app:layout_constraintTop_toBottomOf="@+id/numberField"
         tools:ignore="MissingConstraints" />
     <TextView
         android:id="@+id/txtResult"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_marginTop="52dp"
         android:textSize="20sp"
         app:layout_constraintEnd_toEndOf="@+id/btnPredict"
         app:layout_constraintStart_toStartOf="@+id/btnPredict"
         app:layout_constraintTop_toBottomOf="@+id/btnPredict" />
 </androidx.constraintlayout.widget.ConstraintLayout>
```

Library yang dibutuhkan dalam aplikasi

```
MainActivity.kt
package com.example.mlregresi

import android.content.res.AssetManager
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import org.tensorflow.lite.Interpreter
import java.io.FileInputStream
import java.nio.MappedByteBuffer
import java.nio.channels.FileChannel
```

Menentukan parameter nilai input dan output

```
MainActivity.kt
 private lateinit var interpreter: Interpreter
 private val mModelPath = "linear.tflite"
 private lateinit var resultText: TextView
 private lateinit var editText: EditText
 private lateinit var checkButton: Button
 override fun onCreate(savedInstanceState: Bundle?) {
     super.onCreate(savedInstanceState)
     setContentView(R.layout.activity_main)
     resultText = findViewById(R.id.txtResult)
     editText = findViewById(R.id.numberField)
     checkButton = findViewById(R.id.btnPredict)
     checkButton.setOnClickListener{
         var result = doInference(editText.text.toString())
         runOnUiThread{
             resultText.text = result.toString()
         }
     initInterpreter()
 }
```

Membuat Interpreter

```
MainActivity.kt

private fun initInterpreter(){
   val options = Interpreter.Options()
   options.setNumThreads(4)
   options.setUseNNAPI(true)
   interpreter = Interpreter(loadModelFile(assets, mModelPath), options)
}
```

Menjalankan Interpreter dengan data input dan output

```
MainActivity.kt

private fun doInference(inputString: String): Float{
   val inputVal = FloatArray(size: 1)
   inputVal[0] = inputString.toFloat()
   val ouput = Array(size: 1) {FloatArray(size: 1)}
   interpreter.run(inputVal, ouput)
   return ouput[0][0]
}
```

Membuat fungsi untuk menjalankan model

```
MainActivity.kt
private fun loadModelFile(assetManager: AssetManager, modelPath: String): MappedByteBuffer{
   val fileDescriptor = assetManager.openFd(modelPath)
   val inputStream = FileInputStream(fileDescriptor.fileDescriptor)
   val fileChannel = inputStream.channel
   val startOffset = fileDescriptor.startOffset
   val declaredLength = fileDescriptor.declaredLength
   return fileChannel.map(FileChannel.MapMode.READ_ONLY, startOffset, declaredLength)
}
```

```
package com.example.mlregresi
import android.content.res.AssetManager
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
{\color{red} \textbf{import}} \  \, \text{and} \\ \textbf{roid.widget.EditText} \\
import android.widget.TextView
import org.tensorflow.lite.Interpreter
import java.io.FileInputStream
import java.nio.MappedByteBuffer
import java.nio.channels.FileChannel
class MainActivity : AppCompatActivity(){
    private lateinit var <u>interpreter</u>: Interpreter
   private val mModelPath = "linear.tflite"
    private lateinit var resultText: TextView
    private lateinit var editText: EditText
    private lateinit var <a href="mailto:checkButton">checkButton</a>: Button
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        \underline{\texttt{resultText}} = \texttt{findViewById}(\texttt{R.id.}\underline{\textit{txtResult}})
        editText = findViewById(R.id.numberField)
        checkButton = findViewById(R.id.btnPredict)
        checkButton.setOnClickListener{
            var result = doInference(editText.text.toString())
            runOnUiThread{
                 resultText.text = result.toString()
        initInterpreter()
    private fun initInterpreter(){
        val options = Interpreter.Options()
        options.setNumThreads(4)
        options.setUseNNAPI(true)
        interpreter = Interpreter(loadModelFile(assets, mModelPath), options)
    private fun doInference(inputString: String): Float{
        val inputVal = FloatArray( size: 1)
        inputVal[0] = inputString.toFloat()
        val ouput = Array( size: 1) {FloatArray( size: 1)}
        interpreter.run(inputVal, ouput)
        return ouput[0][0]
    private fun loadModelFile(assetManager: AssetManager, modelPath: String): MappedByteBuffer{
        val fileDescriptor = assetManager.openFd(modelPath)
        val inputStream = FileInputStream(fileDescriptor.fileDescriptor)
        val fileChannel = inputStream.channel
        val startOffset = fileDescriptor.startOffset
        val declaredLength = fileDescriptor.declaredLength
        return fileChannel.map(FileChannel.MapMode.READ_ONLY, startOffset, declaredLength)
```

