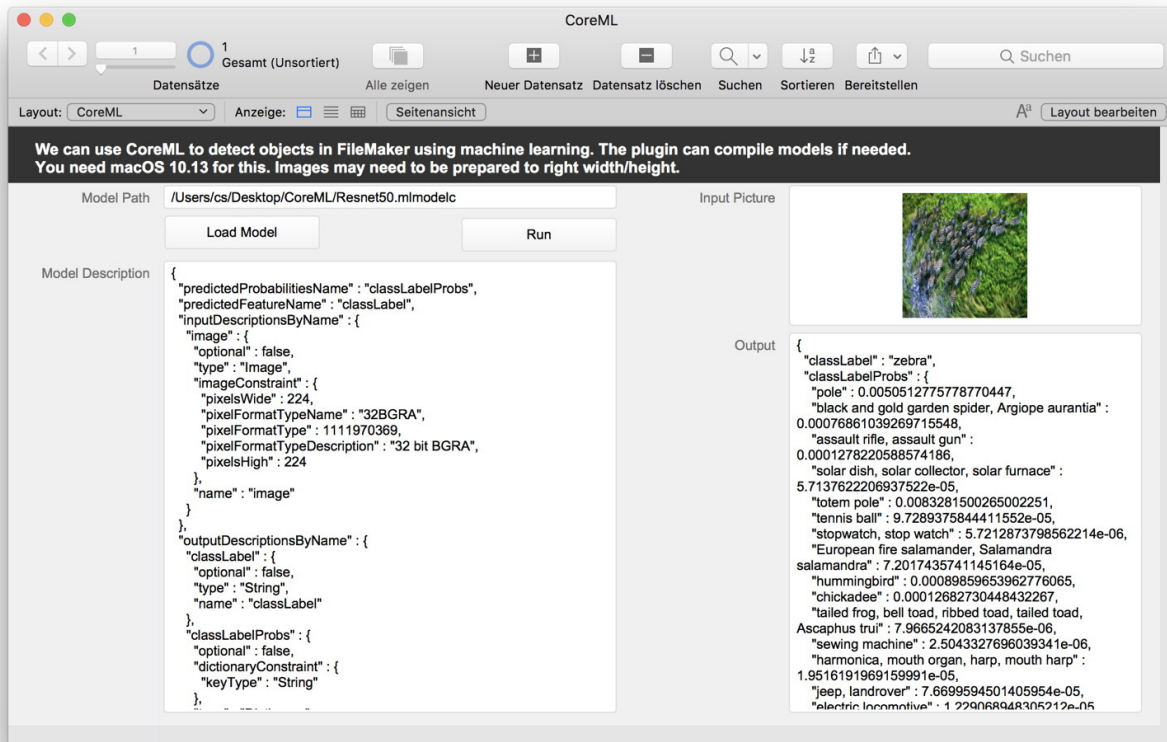


# Train machine learning models on device

For next [MBS FileMaker Plugin](#) we add a new [CoreML](#) functions to update the model on device. If you have an updatable model for CoreML, the Apple framework to use machine learning on Mac and iOS devices (including iPad), then you can now use our [CoreML.Update](#) function to pass new training data to the plugin and we update the model.



You can load a model with [CoreML.OpenModel](#) and you get the description with [CoreML.Description](#), you now see there a new entry trainingInputDescriptionsByName in the JSON. e.g.

```
"trainingInputDescriptionsByName" : {  
  "drawing" : {  
    "optional" : false,  
    "type" : "Image",  
    "imageConstraint" : {  
      "pixelsWide" : 28,  
      "pixelFormatTypeName" : "OneComponent8",  
      "pixelFormatType" : 1278226488,  
      "pixelFormatTypeDescription" : "8 bit one component, black is zero",  
      "pixelsHigh" : 28
```

```

    },
    "name" : "drawing"
  },
  "label" : {
    "optional" : false,
    "type" : "String",
    "name" : "label"
  }
}

```

This shows you there is a drawing parameter for the picture with 28 by 28 pixel resolution in grayscale. The other parameter is the label with the correct output for this image. In a sample call to [CoreML.Update](#) we pass input and output paths for the model files and pass the training data as JSON:

```

MBS( "CoreML.Update";
    "/Users/cs/Desktop/UpdatableDrawingClassifier.mlmodelc";
    "/Users/cs/Desktop/UpdatableDrawingClassifier2.mlmodelc";
    "[{\"drawing\": \"/Users/cs/Desktop/mbslogo.png\", \"label\": \"MBS\"}]" )

```

In the JSON we expect an array of objects. Each object contains the pairs of input and output parameters. Values are passed as numbers, text or objects. For images we decided to allow you to pass the image file as native file path and then the plugin adjust images as needed.

You can build solutions which come with a pre-calculated machine learning model, which is then adjusted on device (e.g. iPad) while the user takes new data and provides correct answers. On the server you can take a basic model to recognize some data and then adjust with all the records you have in your database.

If you are interested to use this functions, please try the 9.6pr3 release or newer. This functionality is available on MacOS 10.15 or iOS 13. Calculation happens on device using GPU if available.

See also [Presentation about a Core ML database for image detection.](#)