##### MACHINE LEARNING MODEL DEPLOYMENT WITH IBM CLOUD WATSON STUDIO

Testing the Deployment:

Once your model is deployed, rigorous testing is essential. You can perform testing within Watson

Studio to ensure that the deployed model works as expected. This testing phase helps verify the

accuracy and reliability of your model's predictions.

Monitoring and Management:

IBM Watson Studio provides tools for monitoring the performance of your deployed model. You

can track its usage, assess its responsiveness, and detect any anomalies. Regular monitoring

ensures that your model continues to provide high-quality results.

Feedback and Iteration:

Collect feedback from users and systems that interact with your deployed model. Utilize this

feedback to iteratively improve your model. You may need to retrain the model with updated data

or adjust its parameters based on user insights.

Version Control:

Maintain version control for your model. Watson Studio allows you to manage multiple versions of

your model, making it easier to track changes and revert to previous versions if necessary.

Scaling and Resource Management:

As demand for your application or service grows, you might need to scale the deployment. IBM

Cloud offers resource management features, allowing you to allocate more computing resources

to your deployed model to handle increased workloads.

Integration:

Integrate the scoring endpoint of your deployed model into your application or system. This

integration enables real-time predictions by sending data to the model's APl endpoint.

Security and Access Control:

Ensure the security of your deployment. IBM Cloud provides features for access control,

authentication, and encryption to safeguard your model and data from unauthorized access and

breaches.

Documentation and Knowledge Sharing:

Document the entire deployment process, including configurations and any challenges faced. This

documentation is valuable for your team and for future reference, ensuring that others can

understand and replicate the deployment.

Collaboration:

If you're working on the deployment with a team, take advantage of Watson Studio's collaboration

features. Share notebooks, data, and insights, and collaborate efficiently to enhance the

deployment.

Performance Optimization:

Continuously assess the performance of your model. Explore opportunities to optimize it, which

might involve hyperparameter tuning, retraining with fresh data, or implementing more efficient

algorithms.

Deploying to an API to watson ML:

Deploy your machine learning model. Watson Studio will provide you with an endpoint URL that

you can use to interact with your deployed model .

wmlLcredentials={

"apikey": "\*\*\*ktttt\*\*

"instance jd": "\*\*\*\*\*\*\*\*\*

"url": "\*\*\*\*\*\*\*tttt\*

client = WatsonMachineLearningAPIClient(wml\_credentials)

Training ML model:

Utilize the machine learning libraries and framework available in Watson studio to train your

model. Make sure to split your data into training and testing set to assets model accuracy.

python

logreg = LogisticRegression(max\_iter=300)

LogisticRegression(C=1.0, class\_weight=None, dual=False, fit\_intercept=True,

intercept\_scaling=1, max\_iter=300, multi\_class=warn',

n jobs=None, penalty=12, random\_state=None, solver='warn,

python

y\_pred = logreg.predict(X\_test)

print(Accuracy of logistic regression classifier on test set: {.2f).format(logreg.score(X\_test,

Y\_test))

Accuracy of logistic regression classifier on test set: 0.77

"python

conf\_matrix = confusion \_matrix(Y\_test, y\_pred)

print(conf \_matrix)

[1209 346]

[144 411]]

`python

print(classification \_report(y\_test, y\_pred))

precision recall f1-score support

00.83 0.75 0.79 1064

10.78 0.85 0.81 1101

micro avg 0.80 0.80 0.80 2165

macro avg 0.80 0.80 0.80 2165

weighted avg 0.80 0.80 0.80 2165

INDEX HTML :

<!DOCTYPE html>

<html lang="en">

<head>

<meta http-equiv="content-type" content="text/html; charset=UTF-8">

<meta charset="utf-8">

<title>IBM Watson Machine Learning Heart Risk Example</title>

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1">

<meta name="description" content="Watson Natural Language Classifier for E-mail Spam" />

<link href="//netdna.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css" rel="stylesheet">

<!--[if lt IE 9]>

<script src="//html5shim.googlecode.com/svn/trunk/html5.js"></script>

<![endif]-->

<link href="//netdna.bootstrapcdn.com/font-awesome/4.4.0/css/font-awesome.min.css" type="text/css" rel="stylesheet">

<!--<link href="stylesheets/watson-bootstrap-dark.css" rel="stylesheet">-->

<link href="stylesheets/style.css" type="text/css" rel="stylesheet">

<body>

<div class="container">

<div class="row header">

<div class="col-lg-12">

<div class="page-header">

<h1>IBM Watson Machine Learning Heart Risk Example

<small>By the IBM Developer Advocacy team</small>

</h1>

</div>

<div>

<p>

Use <a target="\_blank" href="https://www.ibm.com/cloud/machine-learning/">Watson Machine Learning</a> to predict heart risk. Enter information below and see how it is classified.

</p>

<p>

The Training Data: The data set used in this example was mocked up and is only for demonstrative and illustrative purposes only and does not constitute an offering that has gone through regulatory review.

</p>

<p>

<a target="\_blank" href="https://github.com/IBM/predictive-model-on-watson-ml"> Fork the code in GitHub </a> |

<a target="\_blank" href="https://www.ibm.com/cloud/machine-learning/"> WML Product Page </a> |

<a target="\_blank" href="https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/wml-ai.html"> View the docs </a>

</p>

</div>

</div>

</div>

<div class="row">

<div class="col-lg-12">

<div class="well">

<form id="form" role="form">

<div class="input-group classifier-input-form">

<table>

<tr>

<td>

<div>

<label for="age">Age</label>

<input id="age" class="form-control input-lg" placeholder="Enter age" type="text" autofocus required />

</div>

<div>

<label for="gender">Gender</label>

<select id="gender" class="form-control input-lg" placeholder="Enter gender" type="text" autofocus required>

<option value="M">Male</option>

<option value="F">Female</option>

</select>

</div>

<div>

<label for="familyhistory">Family History</label>

<select id="familyhistory" class="form-control input-lg" placeholder="Enter family history" type="text" autofocus required>

<option value="Y">Yes</option>

<option value="N">No</option>

</select>

</div>

<div>

<label for="smoker">Smoker</label>

<select id="smoker" class="form-control input-lg" placeholder="Enter smoker" type="text" autofocus required>

<option value="Y">Yes</option>

<option value="N">No</option>

</select>

</div>

<div>

<label for="exercise">Exercise (minutes per week)</label>

<input id="exercise" class="form-control input-lg" placeholder="Enter exercise" type="text" autofocus required />

</div>

</td>

<td>

<div>

<label for="cholesterol">Cholesterol</label>

<input id="cholesterol" class="form-control input-lg" placeholder="Enter cholesterol" type="text" autofocus required />

</div>

<div>

<label for="bmi">Body Mass Index (BMI)</label>

<input id="bmi" class="form-control input-lg" placeholder="Enter BMI" type="text" autofocus required />

</div>

<div>

<label for="heartbeats">Average heartbeats per minute</label>

<input id="heartbeats" class="form-control input-lg" placeholder="Enter heartbeats" type="text" autofocus required />

</div>

<div>

<label for="palpitations">Number of palpitations per day</label>

<input id="palpitations" class="form-control input-lg" placeholder="Enter palpitations" type="text" autofocus required />

</div>

</td>

</tr>

</table>

<div>

<span class="input-group-btn">

<button class="btn btn-lg classify-btn" type="submit">Classify

<i class="loading fa fa-spinner fa-spin"></i>

</button>

</span>

</div>

</form>

</div>

</div>

</div>

<div class="row answers">

<div class="col-lg-12">

<div class="well">

<h1 class="answer"></h1>

<p class="risk"></p>

<p class="prediction"></p>

<p class="probability"></p>

</div>

</div>

</div>

</div>

<script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.1.4/jquery.min.js"></script>

<script src="//netdna.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script>

<script src="javascripts/index.js"></script>

</body>

STYLE CSS :

.container {

margin-right: auto;

margin-left: auto;

padding-left: 15px;

padding-right: 15px;

}

@media (min-width: 992px) {

.container {

width: 970px;

}

}

.header {

background: #F2FAFC;

border-radius:none;

border-bottom:1px solid #00B2EF;

padding-top: 10px;

padding-bottom:30px;

margin:0 0 30px 0;

}

.page-header {

padding-bottom: 8px;

margin: 36px 0 18px;

border-bottom: 1px solid #eeeeee;

}

.header h1 {

color: #00B2EF;

font-size: 50px;

}

.input-lg {

height: 44px;

}

.btn-lg {

border: solid 1px;

}

table {

max-width: 100%;

background-color: transparent;

}

th {

text-align: left;

}

.table {

width: 100%;

margin-bottom: 18px;

}

td {

padding: 8px;

line-height: 1.428571429;

vertical-align: top;

border-top: 1px solid #eeeeee;

}

.row {

padding-bottom: 20px;

margin-left: -15px;

margin-right: -15px;

}

.search-dropdown-menu {

left: inherit;

position: absolute;

top: 100%;

right: 0 !important;

z-index: 1000;

display: none;

float: left;

width: 800px;

padding: 5px 0;

margin: 2px 0 0;

font-size: 14px;

text-align: left;

list-style: none;

background-color: #fff;

-webkit-background-clip: padding-box;

background-clip: padding-box;

border: 1px solid #ccc;

border: 1px solid rgba(0,0,0,.15);

border-radius: 4px;

-webkit-box-shadow: 0 6px 12px rgba(0,0,0,.175);

box-shadow: 0 6px 12px rgba(0,0,0,.175);

}

.search-dropdown-menu li {

border-bottom: 1px solid #667

}

.search-dropdown-menu li, .search-dropdown-menu li a {

overflow-wrap: break-word;

word-wrap: break-word;

white-space: normal;

}

.jumbotron {

padding-top: 30px;

padding-bottom: 30px;

margin-bottom: 30px;

padding-left: 20px;

color: inherit;

background-color: #eee;

}

@media screen and (min-width: 768px) {

.container .jumbotron, .container-fluid .jumbotron {

/\* padding-right: 60px; \*/

padding-left: 20px;

}

}

INDEX JS :

(function() {

$('.answers').hide();

$('.loading').hide();

$('#form').submit(onFormSubmit);

$('#age').val('40');

$('#gender').val('M');

$('#familyhistory').val('N');

$('#smoker').val('N');

$('#exercise').val('60');

$('#cholesterol').val('10');

$('#bmi').val('20');

$('#heartbeats').val('60');

$('#palpitations').val('1');

function onFormSubmit() {

var age = $('#age').val();

var gender = $('#gender').val();

var familyhistory = $('#familyhistory').val();

var smoker = $('#smoker').val();

var exercise = $('#exercise').val();

var cholesterol = $('#cholesterol').val();

var bmi = $('#bmi').val();

var heartbeats = $('#heartbeats').val();

var palpitations = $('#palpitations').val();

$('.loading').show();

$('.answers').hide();

$('.classify-btn').prop('disabled', true);

$.post("/classify", {age: age,

gender: gender,

familyhistory: familyhistory,

smoker: smoker,

exercise: exercise,

cholesterol: cholesterol,

bmi: bmi,

heartbeats: heartbeats,

palpitations: palpitations

}, function(data) {

renderAnswer(data)

}).fail(function(err) {

renderAnswer(err);

});

return false;

}

function renderAnswer(parsedResponse) {

console.log(parsedResponse);

if (parsedResponse.errors) {

$('.answer').html('Something went wrong :-( ' + parsedResponse.errors[0].message);

} else {

var data = parsedResponse.predictions[0].values[0]

var risk = data[0]

var prediction = data[1][0]

var probability = data[1][1]

$('.risk').html('Heart Risk: '+risk);

$('.prediction').html('Prediction: '+Math.floor(prediction\*100 ).toFixed(0)+'%');

$('.probability').html('Probability: '+Math.floor(probability\*100 ).toFixed(0)+'%');

}

$('.classify-btn').prop('disabled', false);

$('.answers').show();

$('.loading').hide();

}

}());

APP JS :

/\*\*

\* Copyright 2017 IBM Corp. All Rights Reserved.

\*

\* Licensed under the Apache License, Version 2.0 (the "License");

\* you may not use this file except in compliance with the License.

\* You may obtain a copy of the License at

\*

\* http://www.apache.org/licenses/LICENSE-2.0

\*

\* Unless required by applicable law or agreed to in writing, software

\* distributed under the License is distributed on an "AS IS" BASIS,

\* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

\* See the License for the specific language governing permissions and

\* limitations under the License.

\*/

/jslint node: true/

/\*jslint es6 \*/

"use strict";

var express = require('express');

var path = require('path');

var favicon = require('serve-favicon');

var logger = require('morgan');

var cookieParser = require('cookie-parser');

var bodyParser = require('body-parser');

var classify = require('./routes/classify');

var app = express();

app.use(favicon(\_\_dirname + '/public/favicon.ico'));

app.use(logger('dev'));

app.use(bodyParser.json());

app.use(bodyParser.urlencoded({ extended: false }));

app.use(cookieParser());

app.use(express.static(path.join(\_\_dirname, 'public')));

app.use('/classify', classify);

// catch 404 and forward to error handler

app.use(function(req, res, next) {

var err = new Error('Not Found');

err.status = 404;

next(err);

});

// development error handler

// will print stacktrace

if (app.get('env') === 'development') {

app.use(function(err, req, res, next) {

res.status(err.status || 500);

res.json({

message: err.message,

error: err

});

});

}

// production error handler

// no stacktraces leaked to user

app.use(function(err, req, res, next) {

res.status(err.status || 500);

res.json({

message: err.message,

error: {}

});

});

const port = process.env.PORT || process.env.VCAP\_APP\_PORT || 3000;

app.listen(port, function () {

console.log("Server running on port: %d", port);

});

module.exports = app;

DEPLOYMENT URL :

# Replace the credentials here with your own.

# Rename this file to .env before running 'npm start'.

API\_KEY=<API\_KEY>

DEPLOYMENT\_URL=<DEPLOYMENT\_URL>





