Microsoft

Azure SQL R Demo

Last updated: 11/2/2017

# Introduction

First Response Online (FRO) is an application which monitors and manages incidents in a specific city. As manager, you have a set of public resources fire truck, polices and ambulances

In this demo we are going to increase the functionality of this application adding a predictive model which helps to find some hot zones in a city (zones with a high rate of incidents) in order to concentrate the resources according with the risk.

The experiment created will try to predict the number of incidents depending on number of events in near zones and weather conditions.

Before continuing, please ensure you have installed the following:

* A web browser
* VisualStudio
* Microsoft SQL Management Studio

# Experiment

Seattle has registers from incidents happened in the past. We analyze those data and insert into a stored in an Azure SQL Database.

Using the Azure SQL Database capabilities to execute R code in its engine, we have created a R script that trains a model with the stored information. This R script will be executed using a stored procedure. So, if a user wants to predict the occupation for a particular date and conditions, he won’t need to have any knowledge about how R works. Just executing the stored procedure will retrieve the desired results.

# SETUP

| Screen | Click Steps | Demo Script |
| --- | --- | --- |
|  | 1. Open Powershell. | Before starting the demo, it is necessary to follow a couple of steps to deploy the Azure resources. |
| cd {PATH\_TO\_SOLUTION}\src\R demo\FirstResponse.Deploy\Deploy.ps1 | 1. Replace {PATH\_TO\_SOLUTION} with the path where the Demo solution is located. 2. Execute the command. |  |
| .\Deploy.ps1 | 1. Execute the command |  |
|  | 1. Introduce your credentials and then click on Sign in. |  |
|  | 1. Open Visual studio 2. Open MS.Corp.FirstResponse.RDemo solution |  |
|  | 1. Select FirstResponse.Database project and click into Schema Compare |  |
|  | 1. Configure source and destination 2. Click into Compare | By default, the credentials are:   * + User: drwho   + Password: Password!   + Database Name: Seattle911 |
|  | 1. Wait until resolve the update |  |
|  | 1. Click into **Update** button |  |
|  | 1. Select FirstResponse.PopulateDb as startup project |  |
|  | 1. Check App.config file |  |
|  | 1. Execute program 2. Wait until inserts data into database | After that you have data and the procedures for train and predict the model |

# TRAINING THE MODEL

| Screen | Click Steps | Demo Script |
| --- | --- | --- |
|  | 1. Open SQL Management Studio 2. Login into your database |  |
| EXEC [dbo].[TrainIncidencesModel] | 1. Execute Train generator model stored procedure | This will create and serialize into dbo.Model table the DForest model generated from input dataset data |

# Predict

| Screen | Click Steps | Demo Script |
| --- | --- | --- |
| EXEC [dbo].[PredictNumberOfIncidents] | 1. Using this procedure will change | The parameters are:   * Distance1000: number of events in 1 kilometer area * Distance3000: number of events in 3 kilometers area * Distance6000: number of events in 6 kilometers area * Distance6000Plus: number of events in more than 6 kilometers * Temperature: temperature in Celsius degree * Pressure: Sea level pressure * Winspeed: windspeed * Rain: number of cm3 in one hour. |
| EXEC [dbo].[PredictNumberOfIncidents] @distance1000 = 4, @distance3000 = 4, @distance6000 = 4, @distance6000Plus = 9, @temperature = 16, @pressure = 1018, @windspeed = 23, @rain = 1.3 | 1. An example of this execution could be 2. Wait until resolve the number of predicted incidents |  |
|  | 1. The prediction model returns you 1,92549438380448. This means that predict almost 2 incidents. |  |

Take care, if the number is too high maybe it is better to stay at home. That’s and advice for your healthy! 😉