**Big Data**

**Inlämningsuppgift 3 - Analysera med hjälp av Machine Learning**

Började med RainPrediction delen. Själva skapandet av modellen gjorde jag enligt guiden ihop med klassen och gjorde inga ändringar i den.

Jag valde att göra prediction på datan genom Azure function. Varje gång ett meddelande inkommer till min hub görs en prediction på det.



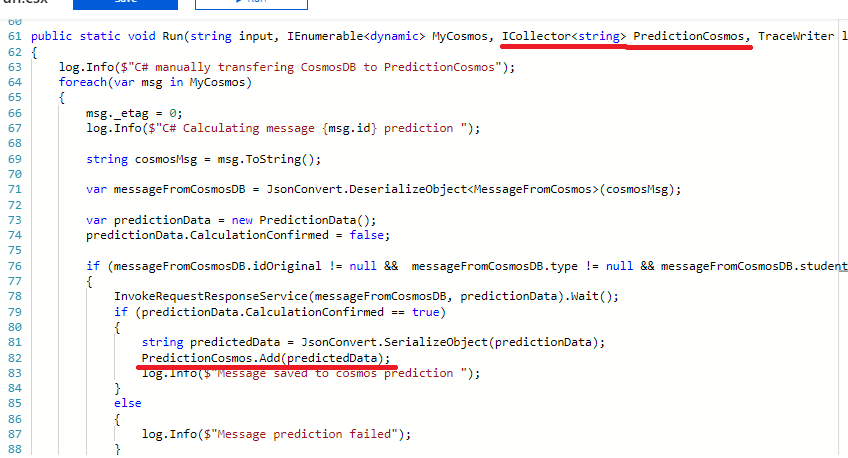
Skapade först objektet jag kommer spara min prediction i. Lagt till en bool för att bekräfta att meddelandet räknats ut korrekt. Denna borde ligga utanför objekten egentligen, finns ingen poäng att spara ner den i en databas.

I requesten skickar jag in temperatur och luftfuktighet som jag plockat ut från inkomna meddelandet via JsonConvert.DeserializeObject. Svaret sparas PredictionData.

Om allt gått bra converteras det till string för att sparas i CosmosDB i egen container för predictions.

Innan jag startade min funktion som ska gå hela tiden gjorde jag även en flytt av tidigare sparade värden från min cosmosDB. Funktionen i sig var den samma men istället för en IotHub-trigger körde jag en manuell trigger som hade min CosmosDB-MessagesFromClass som input, loopade igen med en For Each. Körde varje rad genom request för att sen skicka till CosmosDB-Predictions.

Hade problem att enbart variabeln för out uppdaterades i forloopen och inget sparades förrän funktionen var klar. Dvs enbart det sista värdet av alla 20.000 rader sparades. Höll på länge med detta som du vet och hjälpte med. Provade olika varianter av loopar och async funktioner utan resultat. Hittade tillslut funktionen IColletor <https://docs.microsoft.com/en-us/sandbox/functions-recipes/queue-storage#using-icollector-with-azure-storage-queue-bindings> och då var det precis så enkelt som det borde vara.



## TemperatureAlert

Gjorde en ny device (Weather Station).

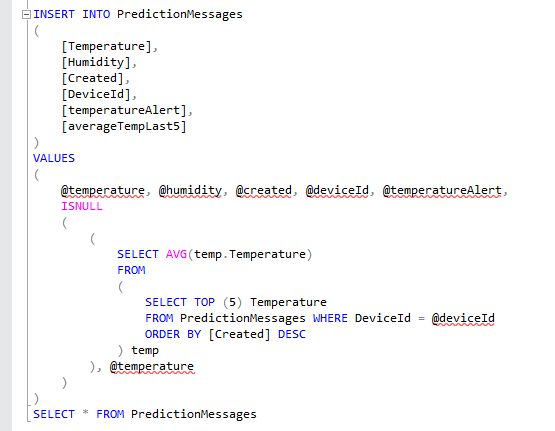
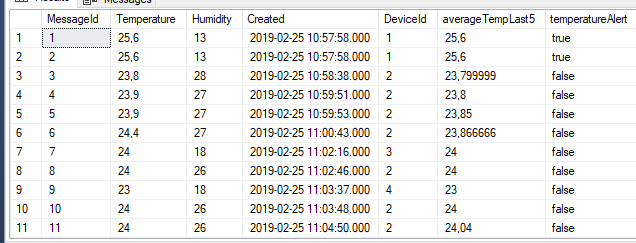
La till parameter ventilation som går på valvePosition.

root["ventilation"] = valvePosition ? "On" : "Off";

La till property för ”alertTemperatrue”.

Map\_Add(properties, "temperatureAlert ", temperatureAlert ? "true" : "false");

Har inte lyckats eller hunnit göra hela uppgiften helt klar. Lagt mycket tid på att skapa en ny SQL databas där jag läst in alla meddelanden som sparats i cosmosDB samt alla nya som inkommer. Gjorde då en ny tabell i min DB som heter PredictionMessages genom en azureFunction (eller 2 funktioner, en som hämtat tidigare sparade samt en som tar emot the nya kontinuerligt). När meddelandet läggs in kan jag då kolla om temperaturen är över t.ex. 25 och i fall registreras ”temperatureAlert” som true, räknar även ut medelvärdet från just den devicen ifrån de senaste 5 meddelanden för att lättare kunna räkna ut en prognos.



I machinelearning har jag då läst in min PredictionMessages som dataset, behövde convertera en del för att få den accepterad. ML har inte decimal som giltigt format så fick convertera dessa till float. Splitade datan för att få en predict, modellen tränades då för att räkna ut om en temperatureAlert skulle triggas utifrån

* Temperature
* Humidity
* Created
* AvgTemp

Problemet är att jag får en realtidsprediktion snarare än en framtida. Härifrån om det hade fungerat korrekt var min plan att i min tidigare azurefunction som läser in ny meddelande för att skicka till prediction att jag skulle använt min egna modell. Ifall utfallet då hade varit att temperaturen sannolikt skulle utlösas skulle jag i samma funktion då skicka tillbaka till den devicen genom devicemethodcallback en funktion som då öppnade ventilationen.