## Learning Outcomes and Prerequisites

**In this module, you will:**

* Conduct an end-to-end explainability exercise, including model-building, evaluation, and other considerations pertaining to explainability
* Apply explainability algorithms to create interpretable models

**To be successful in this module, prior knowledge is recommended in :**

* Python
* Module 1: The Big Picture of Trustworthy and Explainable AI
* Module 2: AI Explainability 360 for Explainable AI

## Overview of the Data Set

The [NHANES CDC questionnaire datasets](https://wwwn.cdc.gov/nchs/nhanes/search/datapage.aspx?Component=Questionnaire&CycleBeginYear=2013) are surveys conducted by the organization involving thousands of civilians about various facets of their daily lives. There are 44 questionnaires that collect data about income, occupation, health, early childhood and many other behavioral and lifestyle aspects of individuals living in the US. These questionnaires are thus a rich source of information indicative of the quality of life of many civilians.

Click <https://wwwn.cdc.gov/nchs/nhanes/search/datapage.aspx?Component=Questionnaire&CycleBeginYear=2013> link to open resource.

## Tutorial Notebook - Run this on Watson Studio

Click [tutorial\_notebook.ipynb](https://learn.ibm.com/pluginfile.php/1562507/mod_resource/content/6/tutorial_notebook.ipynb?forcedownload=1) link to download the file.

## Exercise - Try This Out!

Complete this exercise and observe and record the results, then compare your results to the provided solutions document.

## Module 3 Summary

**1**. AI Explainability 360 (AIX360) includes many different algorithms capturing many ways of explaining, which may result in a daunting problem of selecting the right one for a given application. We provide some guidance to help. The following decision tree will help you in selecting the right one according to your usecase.  
F,{232666ce-ef77-464d-8deb-d4c3a8d6aef6}{116},13,8.208333

**2.** In this tutorial we learnt how a CDC questionaire answered by thousands of individuals could be **summarized** by looking at answers given by a few prototypical usersusing **Protodash**. Here the Protodash generated prototypical users using only the features(X's) of the dataset.

**3.** In this tutorial we learnt how how **SHAP can explain the prediction of an instance x by computing the contribution of each feature to the prediction**. Here SHAP uses both features(X's) and labels(Y's)

**4.** In this tutorial we learnt how easy it is to use the **TED\_CartesianExplainer** if you have a training dataset that contains explanations. The framework is general in that it can use any classification technique that follows the fit/predict paradigm, so that if you already have a favorite algorithm, you can use it with the TED framework.

The main advantage of this algorithm is that the quality of the explanations produced are exactly the same quality as those that the algorithm is trained on. Thus, if you teach (train) the system well with good training data and good explanations, you will get good explanations out in a language you should understand.

The downside of this approach is that someone needs to create explanations. This should be straightforward when a domain expert is creating the initial training data: if they decide a loan should be rejected, they should know why, and if they do not, it may not be a good decision.