|  |  |
| --- | --- |
|  |  |



**[Document title]**

**FedEx Proprietary and Confidential**

**Printed Copies for Reference Only**

Prepared by: UMRF

Manager: Teena D. Cummings

Last Revision: [Date]

**Table of Contents**

[1. Document Information and Revision History 3](#_Toc531255088)

[2. Introduction 4](#_Toc531255089)

[2.1 Overview 4](#_Toc531255090)

[3. Scope ……………………………………………………………………………………………… 4](#_Toc531255091)

[3.1 Scope Definition 4](#_Toc531255092)

[4. [Content] 4](#_Toc531255093)

[5. Reference Materials: 4](#_Toc531255094)

# Document Information and Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author(s)** | **Revision Notes** |
| 0.1 | [Date] | [Name and ID] | First Draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Overview

[Overview]

# Scope

## Scope Definition

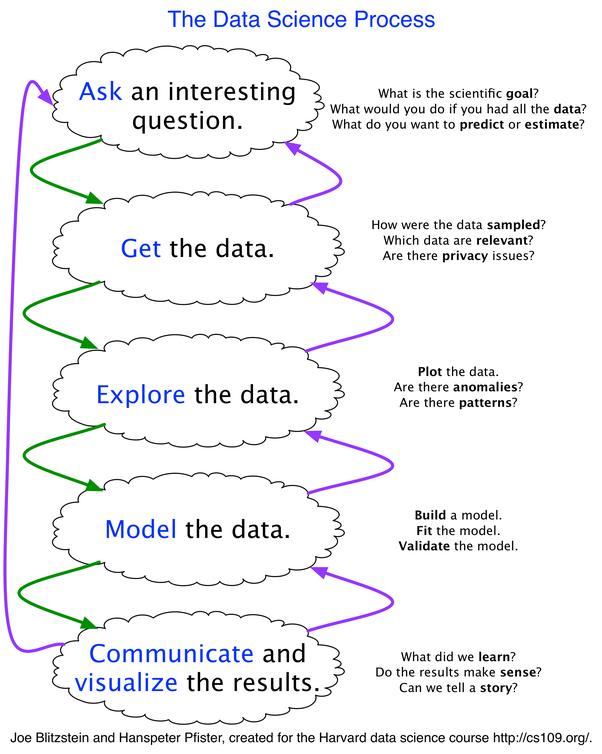
# Data Analysis

Terminology: In machine learning, mathematics, and other fields, the same concept can be referred by different terms. For example, an observation, or instance, corresponds to an n-dimensional feature vector, where n is the number of features in the dataset. These represent an individual object, observation, problem, etc. that has the features described in the dataset. A feature of a dataset can also be referred to as an attribute, field, or variable. These are all the same thing. They all correspond to an individual measurable characteristic of the observed phenomenon (Age, Temperature, Time, Category, etc). Each feature will have its own data type (numeric, nominal, or ordinal). Generally, these will be the columns in a dataset. Generally, these will be the rows in a dataset. In supervised learning problems, features have another important aspect. There will generally be one class, or category, (of any data type) that we want to learn/predict. While the class could be considered one of the features of the dataset, it will be treated differently especially at the time of model building. In mathematics and traditional science, the class corresponds to the dependent variable. The independent variable(s), then, are the n-1 features that are not the class to be predicted. For ease of understanding, in this document, the terms feature, observation, and class will be used to refer to their respective concepts.

Disclaimer: This is a very general guide and each individual problem will have its own challenges and need to be tackled in a way to specific to the datasets of interest. For example, practices such as dropping attributes with varying degrees of NA values may not be appropriate for other projects and it may be preferable to find another way to deal with that issue.

This document will give an overview as to an example workflow for tackling a data science problem. Any technical examples will be using python. Familiarity with python, R, or at least some programming language is highly recommended, however, there will be some tutorials in this guide.

Valuable reference for workflow at: <https://towardsdatascience.com/a-data-science-workflow-26c3f05a010e>

It is very important to first note what the actual problem that needs to be solved is. If this cannot be done, such as in the example project, it may be sufficient to just explore the data, see what patterns emerge, and what kinds of models can be built.

Note that data science work flow is non-linear, iterative, and cyclical.

## Data Exploration / Cleaning

* + 1. *Introduction*

Real world data is often very messy. Most of the time spent on these types of project is spent on exploring and cleaning the data. Using either python’s pandas module or a spreadsheet application to simply view the data is a good first step. Familiarization of all the attributes is important, as well as noting what their data types are and what kind of information may be missing.

* + 1. *Cleaning NAs*

A first step in exploring a dataset with varying degrees of missing data could be to identify those attributes that have all missing or a large number of missing values. These attributes can either be dropped entirely, modified, or left NA depending on the needs of the problem at hand. It is not always possible, but sometimes it’s useful to just eliminate all NAs from a dataset, proceed with further analyses or model building and return to deal with the NA values once more is known. In the example project, attributes containing only NAs were dropped as well as attributes that were comprised of over 10% NAs.

Additionally, even if a feature has no NAs, we want to look for attributes that contain all the same value, or mostly the same value. This can be thresholded as one desires. In the example project, this threshold was chosen to be 80%. If the class to predict is already known (supervised), we obviously don’t want to eliminate the class even in this case, although, as in the example, this would mean that subsampling and/or resampling will be required before model building.

In the case of a supervised learning problem, we should also check that for the class there are no NAs. If there are, these observations may have to be dropped entirely.

* + 1. *Asdfs;dlkfj;j*
    2. *;slkjf;alsjdf*
    3. *Asdf*
    4. *Asdf*
    5. *Asdf*

## Visualization

## Asd;flkjasdf

# Reference Materials:

[References]

----------------------------------END OF THIS DOCUMENT----------------------------------