**Data Science Case Study of Absenteeism**

**To tackle this case study, must have skills:**

1.Able to manage information (will use SQL)

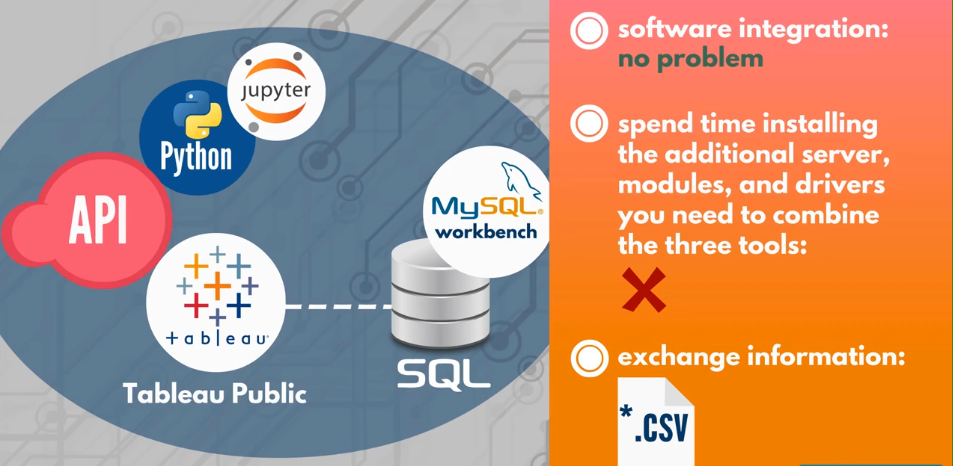
2. Have a substantial amount of mathematical and statistical tools (will use python)

3. Present results in most intuitive way (ill use Tableau)

4. Be able to tackle the problem ffrom businessperspective (self knowlede of feature engineering)

**Table of contents:**

1. Focus on Business Problem
2. Look into the dataset
3. Apply logostic regression
4. Create a Dataviz to provide deeper understanding of results



**Business Tasks/Business Problem:**

This case study will address ABSENTEEISM at a company during work time.

ABSENTEEISM = Absence from work during working hours, resulting in temporary incapacity to execute regular working activity.

**Purpose :** Explore whether a person presenting certain charateristics is expected to be away from work at some point in time or not. Calculated on basis of, how far they live from workplace, how many childrens or pets they have, do they have higher education and so on.

**The problems:**

Higher competetiveness -> incresed pressure

Unachievable business goals -> raised stress levels

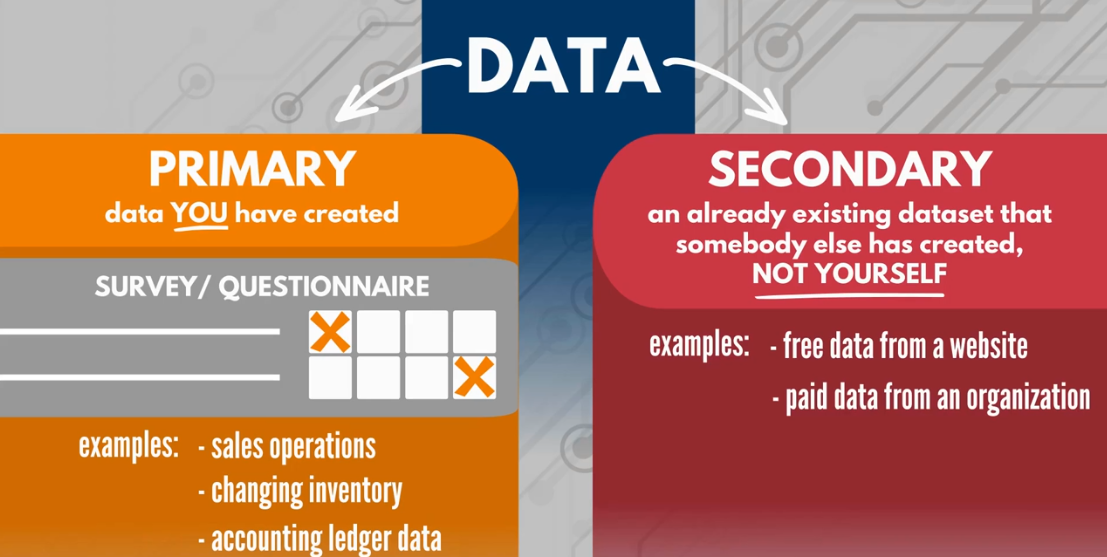
Elevated risk of becoming unemployed -> raised stress levels

**Advantages of this case study:**

It can enhance our descision making, by reorganising the work process of employees in a way that increases employees productivity and hence increase the quality of work in the firm.

**About Data:**

We will be using secondary data which already have information about certain characteristics of employees regarding their absenteeism.



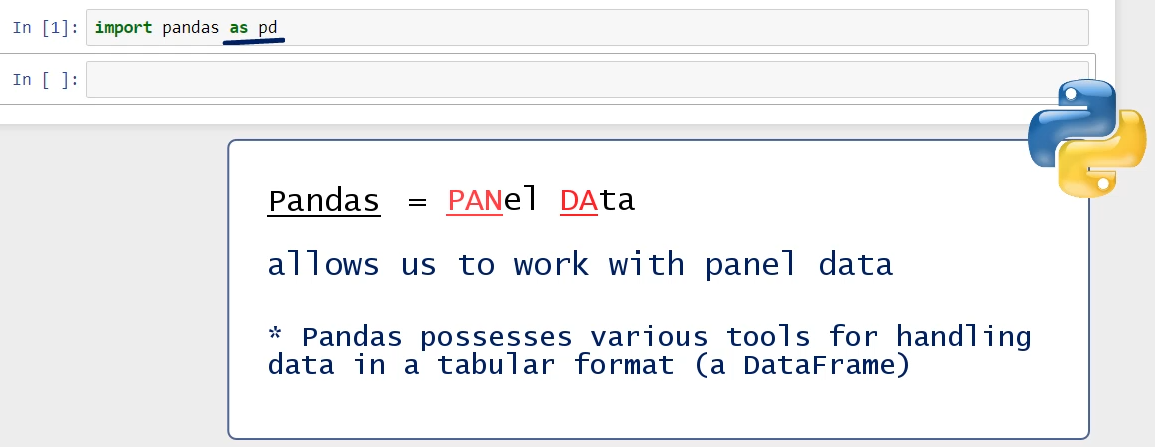
**Data PreProcessing:**

This is a group of operations that will convert the row data of ours (the given secondary data) into a format that is easier to understand and hence useful for urther processing and analysis.

This helps to fix the provlems that can inevitably occur with data gathering and organize the information in a suitable and practical way.

First, we will preprocess the data. We devote a significant amount of time to this step as it is a crucial part of every analytical task.

Will do this in Colab notebook…



A screenshot of a computer

Description automatically generated

A close-up of a box

Description automatically generated

**NOTE:**

**A white grid with black text

Description automatically generated**

**Quick Notes on Regression Analysis:**

A popular tool in data analytics, machine learning, advanced statistics, and econometrics, is regression analysis.

Roughly speaking, this is an equation which on one side has a variable, called a dependent variable, because its value will depend on the values of all variables you see on the other side.

The variables on the right side are all independent, or explanatory. Their role is to explain the value of the dependent variable.

There are more terms that can be used for these variables in the same context. The dependent variable can also be called a target, while the independent variables can be called predictors.

Logistic Regression is a type of a regression model whose dependent variable is binary. That is, the latter can assume one of two values 0 or 1, True or False, Yes or No.

Therefore, considering the values of all our features, we want to be able to predict whether the dependent variable will take the value of 0 or 1.

Apart from logistic regression, there are many other types of equations that allow us to calculate the dependent variable in a different way. Logistic regression is just one of them and it is one that has been used massively.

Anyway, you would most often hear professionals say that they are trying to find a regression model, or, simply, find a regression, that has a high predictive power. In other words, what they are trying to do is settle upon an equation that could be used to estimate expected values for the dependent variable with great precision.

**Conclusion points on dataset analysis:**

1. There is no missing value or null value
2. Dependent vairiable/ target -> Absenteeism time in hours
3. All other features/coulumns are independent variables which we can use in our equation for prediction of target.
4. We cannot include ID column as it will decrease the model’s precision due to being not carrying any numeric information

**Dataset Features description:**

A screenshot of a computer

Description automatically generated

We will be grouping these reason types to get better precesion and avoid multicollinearity

**Note:**

Creating checkpoint while coding is very crucial in models.

Checkpoints are interim save of our work, this is done to have more control over our data.

We needed a change at any instance of model we will have to reexute various logics, so instead we can save a immediate copy of current save of the dataframe by naming it as its current state we are at.

A screenshot of a chat

Description automatically generated

**Part 2: Using logistic regression and other ML methods for developing module**

NOTE:

Creating new notebook for this is recommended, as due to lots of cells previous notebook may start lagging, we can test out the exported csv files with other methods without disturbing previous logic and so on.

**Most common steps for Standardize the data:**

**A screenshot of a computer

Description automatically generated**

The absenteeism\_scaler here will be used to subtract the mean and divide the standard deviation featurewise. This is done when transform() is called on the data

The fit() will calculate and store the mean and std devation in absenteeism\_scaler object.

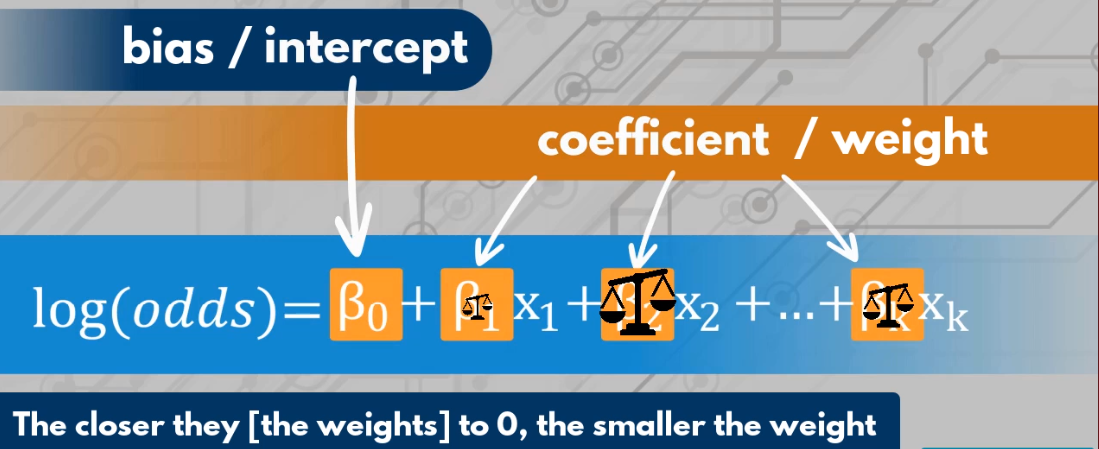
**Splitting the data for train and test:**

We will use sklearn library for this to split the data into train and test using train\_test\_split

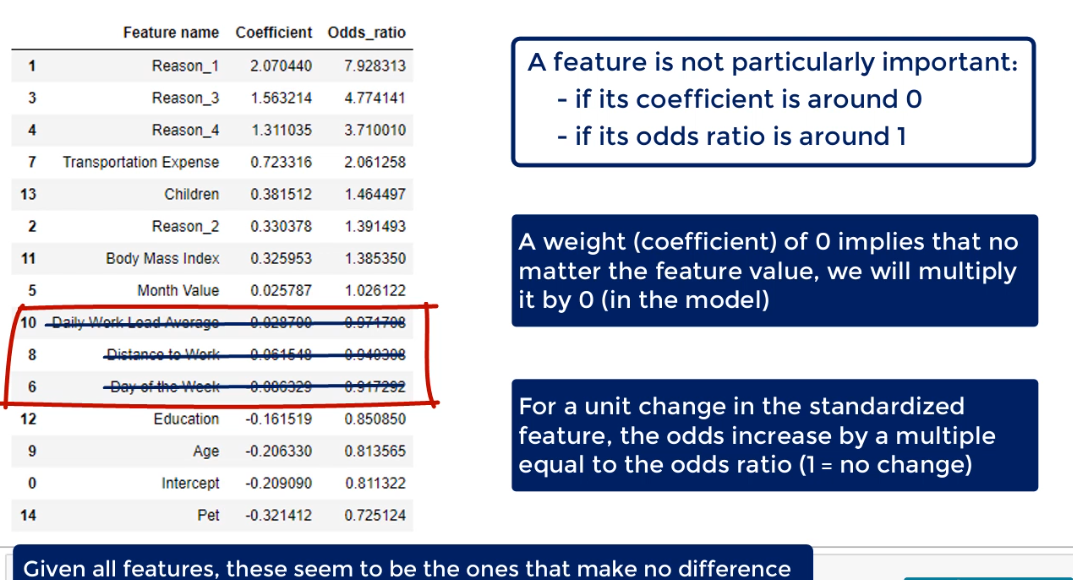
By default shuffle is enabled. By default split size is set to 75% train and 25% test.

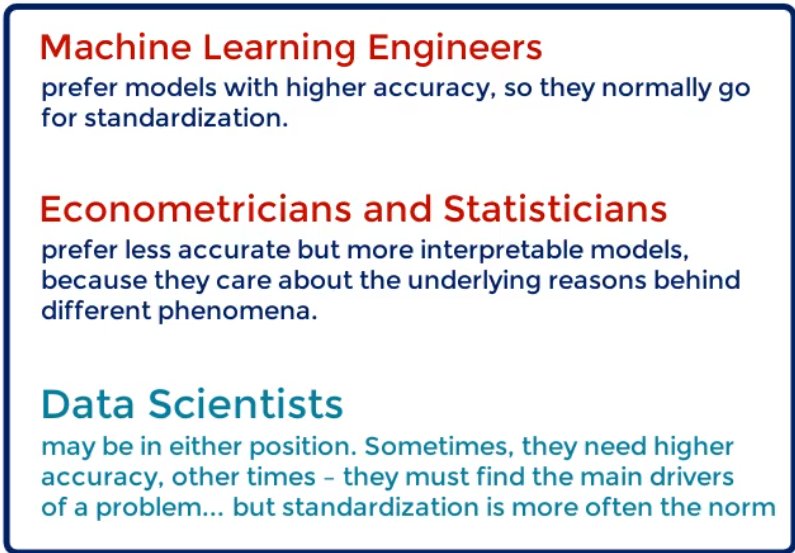
**NOTE:** ML is generally 90% preprocessing and 10% modelling

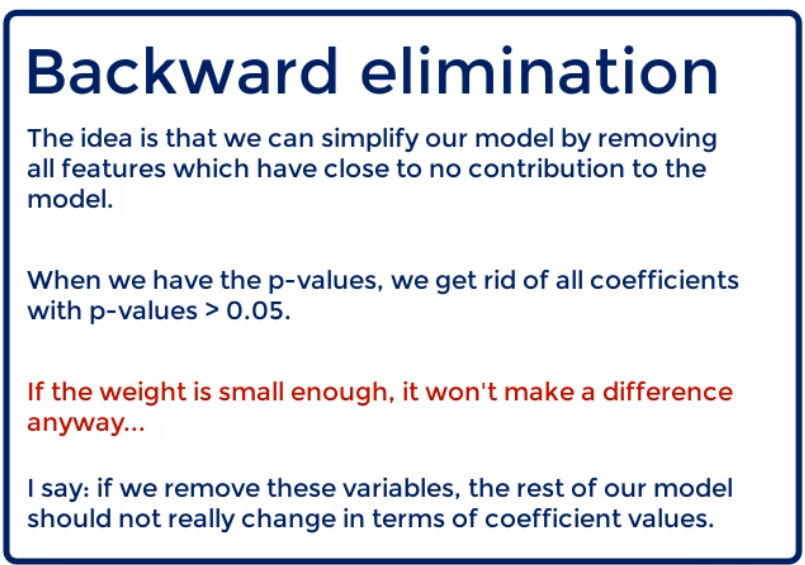
To use the logistic regression model outside of python like maybe in Tableu, we must get our hands on the coefficients and intercepts.



Checking exponentail of the weights to know how much the features are affecting the model









To save a model we can use **pickle** module, this python module is being generally used to convert a python object into a chrater stream.

Pickle is the standard Python tool for serialization and deserialization. In simple words, pickling means: converting a Python object (no matter what) into a string of characters. Logically, unpickling is about converting a string of characters (that has been pickled) into a Python object.

There are some potential issues you should be aware of, though!

1. Pickle and Python version.

Pickling is strictly related to Python version. It is **not recommended**to (de)serialize objects across different Python versions. Logically, if youre working on your own this will never be an issue (unless you upgrade/downgrade your Python version).

1. Pickle is slow**.**

Well, you will barely notice that but for complex structures it may take loads of time to pickle and unpickle.

1. Pickle is not secure.

This is evident from the documentation of pickle, quote: Never unpickle data received from an untrusted or unauthenticated source. The reason is that just about anything can be pickled, so you can easily unpickle malicious code.

A screenshot of a computer program

Description automatically generated

**Deploying the Module**

A close-up of a diagram

Description automatically generated

Simple steps to use the saved module:

Its recommended that the files and all modules are present in same folder.

A screenshot of a computer screen

Description automatically generated

**Visualization of prediction output in Tableau:**

**A screenshot of a computer

Description automatically generated**