In this discussion, you must describe various ways (at least two) to identify the features that are most significant in predicting or classifying the target feature. Provide references and give python code examples (MLO2)

1. Read each question -- there may be more than one.
2. Post your responses in the appropriate threads. Students are required to make their initial post by Thursday at 11:59 p.m. (Pacific Time).
3. Responses should be of sufficient length with proper grammar.
4. Respond to at least one of your classmates' postings. Responses are due by Sunday at 11:59 p.m. (Pacific Time). There is a penalty of loss of points for less than minimum posts. This feature will keep you engaged.
5. Review your postings to see who has responded to you

Two important processes that I believe that are important in predicting/classifying a target feature are finding/collecting/loading the dataset to begin examining the features as well as the Data Analysis process behind identifying features and planning to manipulate the data in a way which maximizes its potential.

Things like read\_sas, read\_sql, or read\_xml could be used from Pandas to load the dataset and bring it into the Python Lab or Notebook platform. From there it is important to understand basic shape and types of data one is dealing with. Data analysis incorporates things like histogram plotting to understand outliers, bar or scatter plots, or correlations matrices to determine covariates.

In terms of feature engineering - normalization is always quite an important step as well as basic examination of the dataset. This code could assist with that process:

import pandas as pd

from sklearn.preprocessing import MinMaxScaler, StandardScaler Happy\_Dataframe = pd.DataFrame({'x':[1,2,3,4,5]}) print("Dtypes:", Happy\_Dataframe.dtypes.to\_dict()) # variable names & classes print("Normalized:", MinMaxScaler().fit\_transform(Happy\_Dataframe).ravel()) print("Standardized:", StandardScaler().fit\_transform(Happy\_Dataframe).ravel())

References:

McKinney, W. (2010). *Data structures for statistical computing in Python*. In S. van der Walt & J. Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (pp. 51–56). <https://doi.org/10.25080/Majora-92bf1922-00a>

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825–2830. [http://jmlr.org/papers/v12/pedregosa11a.html](https://jmlr.org/papers/v12/pedregosa11a.html)