Workflow for Machine Learning Projects
B. MacLennan. 09/19/17

These steps are not strictly linear, and may overlap, but illustrate the general procedure.

1. **Data exploration**. Inspect the file.
   a) What is the file format? Textual? Binary? csv?
   b) Is an index or dictionary available that describes the names and types of the attributes?
   c) What are the types of the attributes or features? Integer? Real? Boolean? Character?
   d) Inspect the numeric attributes. Are there missing values (blank), apparently incorrect values, or illegal values (NAN)?
   e) Inspect the non-numeric attributes for missing or anomalous values.
   f) Document what you have discovered.
2. **Data preparation**: clean, prepare, and standardize the data.
   a) Compute statistics for the attributes (mean, SD, min, max, quartiles, number of values). This will help you identify missing and anomalous values.
   b) Decide on an imputation strategy for missing or incorrect data. Document your reasons.
   c) Apply your imputation strategy.
   d) Convert nominal attributes to numerical.
   e) Standardize numeric attributes where advisable (e.g., z-normalization).
   f) Document all the above.
3. **Dimension reduction.** Consider if dimension reduction is advisable and choose a method. (Alternately, start simple and wait to see if later steps need to be improved by dimension reduction.)
4. **Implement** the algorithm.
   a) Start with an implementation that is quick-and-dirty but obviously correct.
   b) Test it on some made-up data for which you know the answer.
   c) Optionally vectorize your program for better performance.
5. **Separate** real data into training, validation, and test sets.
6. **Train** the model.
   a) For given hyperparameters, train and cross-validate the model.
   b) Compare training and cross-validation error for various hyperparameter values. If you have a regularization or complexity parameter, plot training and CV error against it, and look for the "elbow."
   c) Select hyperparameters that give the best generalization.
7. **Testing**. Test the trained and validated model on previously selected test data.
8. **Interpret** the results. Do they make sense?
9. **Improve** the implementation and training. Now that it's working, there may be things you can do (e.g., vectorization, use of optimized libraries) to improve space/time performance and accuracy.