Materials Informatics – Fall 2020

Computer Project 1

Due on: Oct 10

We are going to use a real material data set to do a simple classification experiment. The data come from the publication

> T. Yonezawa, K. Suzuki, S. Ooki and A. Hashimoto, "The Effect of Chemical Composition and Heat Treatment Conditions on Stacking Fault Energy for Fe-Cr-Ni Austenitic Stainless Steel." *Metall and Mat Trans A* (2013) 44: 5884.

This data set contains the experimentally recorded values of the stacking fault energy (SFE) in austenitic stainless steel specimens with different chemical compositions. The SFE is a microscopic property related to the resistance of austenitic steels. High-SFE steels are less likely to fracture under strain and may be desirable in certain applications. The purpose of the experiment is to develop a model to classify a steel sample as high-SFE or low-SFE based only on the atomic composition.

The data set contains 17 features corresponding to the atomic composition (percentage weight of each atomic element) of 473 steel specimens. We face the problem that the data matrix contains many zero values, which are typically measurements that fell below the sensitivity of the experiment, and are therefore unreliable. These constitute *missing values* (this can occur for other reasons as well, such as a faulty or incomplete experiment). One option to address this issue is to apply *data imputation*. For example, a simple imputation method is to fill in missing values with an average of neighboring values. Given a large sample size and abundance of features, a simpler, and possibly safer, option is to discard measurements containing zero/missing values. Here, we discard all features that do not have at least 60% nonzero values across all sample points, and then remove any remaining sample points that contain zero values. The remaining training points are categorized into high-SFE steels (SFE$\geq$45) vs. low-SFE steels (SFE$\leq$ 35), with points of intermediate SFE value being dropped. This results in a reduced data set containing 123 specimens and 7 features.

Assignment 1: Classification using filter feature selection.

(a) (Data pre-processing.) Study the code in `c01_matex.py` and run it to obtain the filtered data matrix for the experiment. Pick the first 20% of the sample points to be the training data and the remaining 80% to be test data.

(b) Using the function `ttest_ind` from the `scipy.stats` module, apply Welch's two-sample t-test on the *training data*, and produce a table with the predictors, T statistic, and p-value, ordered with largest absolute T statistics at the top.

(c) Pick the top two predictors and design an LDA classifier. Plot the training data with the superimposed LDA decision boundary. Plot the testing data with the sumperimposed previously-obtained LDA decision boundary. Estimate the classification error using the training and test data. What do you observe?

(d) Repeat for the top three, four, and five predictors. Estimate the errors on the training and testing data (there is no need to plot the classifiers). How do the training and testing errors behave?

Assignment 2: Classification using wrapper feature selection.

Here we will consider wrapper feature selection, with the apparent error estimate of the designed LDA classifier as the criterion. You should repeat item (a) of the previous assignment to get the training and testing data.

We will employ two simple feature selection methods: exhaustive search (for 1 to 5 variables) and sequential forward search (for 1 to 5 variables). If two candidate feature sets have the same minimum apparent error, pick the one with the smallest indices (in "dictionary" order): compare the smallest index, if tied, compare the second smallest one, etc. In the sequential forward search, you should continue to add features until the desired size (from 1 to 5), even if doing so does not decrease the apparent error. Therefore, each person will determine 10 feature vectors for each of the 2 classification rules, for a total of 20 feature vectors.

Each person will submit a table with the feature sets found. For each row of the table (variable set found), the corresponding error estimate and the test-set estimate (using the test set provided) should be indicated. Intepret the results. In particular, here are examples of questions you should address:

- How do you compare the results against each other and against the results obtained with the filter feature selection methods in Assignment 1?
- How do you compare the apparent error of the selected feature sets to their test set error?
- How do you compare the feature selection methods based on the feature sets found and the estimates of the true error?
- How do you think the results might change if there were more training points available?