Practical 1: Regression

Antonio (email, Camelot.ai username)
Fangli (email, Camelot.ai username)
Xihan (email, Camelot.ai username)

February 8, 2018

1 Technical Approach

We began by performing exploratory analysis on the sample dataset we were given, which consisted of a training set of 1 million molecules with 256 binary predictors, in addition to their SMILES string and the HOMO-LUMO gap we were seeking to predict. The test set consisted of 824,230 molecules with the same set of predictors as well as their SMILES string. The sample code we were given implemented a default Linear Regression model on the full set of 256 predictors, yielding an $R^2_{\text{LR}} = 0.461(\text{MSE}_{\text{LR}} =?)$ on the full training set, and a default Random Forest regression with $R^2_{\text{RF}} = 0.554(\text{MSE}_{\text{LR}})$.

Initial inspection found that out of 256 molecular features, 221 of them were unexpressed (i.e. had $x_i = 0$) for *all* molecules, both in the training set and the test set. Dropping unimportant features would normally call for K-fold cross-validation across the training set to ensure those features are consistently unimportant in all cases. However, in the case of null values of certain features for every element of the training set, it is not only legitimate but necessary to drop those features from all further analysis, as fitting along null dimensions would constitute a form of overfitting.

Having reduced the sample dataset to 31 expressed molecular features, we performed regularized and non-regularized linear regression with cross-validation[1], under the following methods:

Non-regularized linear regression Yields $R^2 = ...$

Ridge Regression

Lasso

Elastic Net

The results from linear regression on the sample dataset led us into pursuing 2 parallel tracks:

1. Feature engineering: (Fangli)
2. Non-linear methods: (Xihan, Antonio)

[1] After initially trying 3-fold, 5-fold and 10-fold cross-validation, we settled on 5-fold cross-validation as the best compromise between accuracy and computational speed

Mention Features

Feature	Value Set
Mention Head	$\mathcal{V}$
Mention First Word	$\mathcal{V}$
Mention Last Word	$\mathcal{V}$
Word Preceding Mention	$\mathcal{V}$
Word Following Mention	$\mathcal{V}$
# Words in Mention	$\{1, 2, \ldots\}$
Mention Type	$\mathcal{T}$

Table 1: Feature lists are a good way of illustrating problem specific tuning.

Among these we tried 2 broad categories:

(a) Tree based methods / Ensemble methods (Xihan)

(b) Deep learning (Antonio)

How did you tackle the problem? Credit will be given for:

- *Diving deeply into a method (rather than just trying off-the-shelf tools with default settings). This can mean providing mathematical descriptions or pseudo-code.*
- *Making tuning and configuration decisions using thoughtful experimentation. This can mean carefully describing features added or hyperparameters tuned.*
- *Exploring several methods. This can contrasting two approaches or perhaps going beyond those we discussed in class.*

Thoughtfully iterating on approaches is key. If you used existing packages or referred to papers or blogs for ideas, you should cite these in your report.

2 Results

This section should report on the following questions:

- Did you create and submit a set of predictions?
- Did your methods give reasonable performance?

You must have *at least one plot or table* that details the performances of different methods tried. Credit will be given for quantitatively reporting (with clearly labeled and captioned figures and/or tables) on the performance of the methods you tried compared to your baselines.

Model	Acc.
Baseline 1	0.45
Baseline 2	2.59
Model 1	10.59
Model 2	13.42
Model 3	7.49

Table 2: Result tables can compactly illustrate absolute performance, but a plot may be more effective at illustrating a trend.

3 Discussion

End your report by discussing the thought process behind your analysis. This section does not need to be as technical as the others but should summarize why you took the approach that your did. Credit will be given for:

- Explaining the your reasoning for why you seqentially chose to try the approaches you did (i.e. what was it about your initial approach that made you try the next change?).
- Explaining the results. Did the adaptations you tried improve the results? Why or why not? Did you do additional tests to determine if your reasoning was correct?