Quantitative Evaluation and Baseline for Data Science projects

Why evaluation is done in a MSc IS Data Science thesis?

Data Science is about discovering patterns in data useful for making accurate predictions on unseen data. When you do Data Science, you will create something that does something given a piece of data (a prediction usually). You, your supervisors and the examiner want to know how well your creation works. The evaluation creates a clear heaven in which progress is always measurable.

You demonstrate how well your creation works by using an evaluation on *new unseen data containing the correct prediction (label or value)*. This is often *hand labelled gold standard, or golden truth* data. If you *do not have such data, evaluation is nearly impossible. Almost never is it feasible to create such a dataset by yourself in these 3 months that you have.*

Format

The way you present your quantitative evaluation follows common practice, and can easily and nicely be done using scikit-learn and seaborn. You want to demonstrate how good your model works. So (for a classifier, for regression the story is comparable):

- P,R, F1 scores for each class
- The averages (indicate whether it is macro or micro)
- PR curve or curves if you compare multiple models
- Error analysis based on a confusion matrix

An extra dataset

Find at least one extra dataset, next to that set provided by your internship. The best is open source with published results and scores. This will enable you to know what your model should aim for. Even if your model performs poorly on your internship data, but it still in line with the outside data set, you can finish your master thesis with it. Then, an error analysis with cunning hypotheses on the causes of that strange mismatch are in place.

An extra dataset gives you piece of mind. If something goes wrong, it provides a direction to look for the reasons. It is worth your time to find one. For yourself, for your grade and for your examiner. Once you have your implementation in place, running an experiment on that extra dataset is really not that much extra work.

Unsupervised learning

You probably are doing clustering then. Simply giving some clustering quality scores is not enough. We want to see whether the clusters are useful for the intended application, meaning that they correspond to "natural" groups. We also want to see how good the boundaries are. Most often, you will need hand labelled data to assess this.

A/B testing

Of course, an A/B test is also possible in suitable cases. For this, you find other testing and evaluation setups. Your best compass is *to follow a state-of-the-art example* found in a scientific article with many citations and which is published in a top venue.

No golden data?

There is one way to make this work: by providing a *convincing alternative***.** Recall that a data-evaluation as described above is up to a certain point *objective*. Convince yourself that your alternative is that too. It is not acceptable to just make a prototype, a proof of concept or just presenting a methodology.

Starting with a Baseline ML algorithm
In most Data Science theses, you start with a strong but simple baseline. A baseline:

- gives you insight in achievable performance
- gives insight in the types of errors made
- gives ideas on better/different representations of your data and/or selection of features
- provides you a "pipeline" for doing complete ML experiments, including an evaluation that provides insights, "evidence" on trustworthiness of your learned model and (promising) directions for possible improvements

Helpful examples
This blogpost on *simple* baselines is a great start. You need to realize that doing something absolutely non-fancy as a start is an excellent start!

If you need a worked out example, continue with this post by the same author on text classification, accompanied by a well-organized Jupyter notebook. In fact, such a well worked out notebook is really worth the time and energy to make. You can be proud, easily share your results with peers and receive tips and feedback. It shows that *what you are doing makes sense and leads to (OK, sometimes they are small) improvements.* **Small addition to that notebook:** also try an even simpler representation: letter n-grams, for n in 2,3,4. As a way to add a bit of the structure to the text, you may add *word bi or even trigrams* to your tokens (unigrams).

For the evaluation, you can use the ideas stated in this notebook (PCA plotting to see the separateness of the classes), and also these in the seaborn chapter of the Data Science Handbook to get a feeling for the problem *before* you start training a model. Use the ideas in this notebook for inspecting *important features* for all of your classes. Does it make sense? Are you not overfitting?

Checklist for setting up
1. Is the task clear? Did you get some idea on how hard the task is?
2. Is the model simple enough and the results acceptable (i.e. better than majority class)?
3. Are the "obvious" improvements tried out and evaluated?
4. Does the presented model make sense?
    - Are "important" features for the model also acceptable for you?
5. Do the errors make sense? Is it indicated where improvements could or should be made?

Eh, what if I do unsupervised learning?
- Check the treatment of techniques like PCA and Gaussian Mixture models in the Data Science Handbook.
- You will need to evaluate, but if you really have no labels, that seems impossible. So you need at least something.
- If you have data "clustered" in classes, you can see how well your cluster algorithm caught these classes.
- There are also scores which say something about the intrinsic quality of the clustering, like the silhouette score and the calinski_harabasz score.
- Some useful links:
    - https://towardsdatascience.com/how-to-evaluate-unsupervised-learning-models-3aa85bd98aa2 is really introductory
    - https://arxiv.org/pdf/1905.05667.pdf is quite exhaustive and goes quite deep
    - For NLP: https://dl.acm.org/doi/pdf/10.5555/2140458.2140463
    - And as the last one, something special which is not for the *faint-hearted*.