Lab 1 slides

Jacob Rinderud

K-NN

The k-Nearest Neighbors (kNN) classification model is a type of instance-based learning algorithm in machine learning. It operates on the principle that similar things exist in close proximity.

Logistic Regression

Logistic Regression is a **statistical** model used in machine learning for binary classification problems. It uses the logistic function to model the probability of a certain class or event.

- **Pros & Cons**: Logistic regression is simple, fast, and provides probabilities for predictions. However, it may underperform when there are multiple or non-linear decision boundaries. It's not powerful enough to capture more complex relationships.

Decision Tree

A **Decision Tree** is a supervised machine learning model used for classification and regression tasks.

- **Advantages**: Decision Trees are easy to understand and interpret, require relatively little effort for data preparation, and **can handle both numerical and categorical data**.
- **Disadvantages**: They can easily overfit or underfit the dataset, leading to poor predictive performance. They can also be unstable.

Forest

A **Random Forest** is also a supervised machine learning model used for both classification and regression.

- **Advantages**: Random forests are highly accurate and robust. It does not suffer from the overfitting problem.
- **Disadvantages**: The main limitation of the Random Forests algorithm is that a large number of trees can make the algorithm slow for real-time prediction.

Reasoning

I chose to evaluate the k-NN, log-reg, decision tree and random forests since they are relatively simple yet sufficiently different for me to learn something (and comparing them making sense).

Evaluation of the methods

k-NN

This method is nice because of its simplicity but I think it would shine brighter with a multiclass problem. It did not perform well enough for my purposes and this application.

log-reg

The logistic method is not good at capturing a more complex relationship, which I would argue that this application and features are. It performed worse than I hoped and not well enough for this application.

Decision tree

A decision tree is explainable, which is important in a lot of settings, and very simple, both to understand and to use. It can capture more complex relationships but suffers from overfitting/bias. I like this method but as a part of an ensemble model.

Random forest

The random forest is basically a decision tree but without the overfitting/bias problem. I love this model because it is so simple and performant. Even if it is a little slow...

I ultimately chose to go with the random forest classifier since it is an ensemble on trees which have very good characteristics for this application. Being able to handle both numerical and categorical data with minimal data preparation and using the ensemble method minimise the bias and not worrying about overfitting.

What I would do differently

If i did this again, I would do something to combat outliers and preprocess the data more for the first two methods. Some features had categorical nature and might have affected the results of the first two methods.

I was plannig on evaluating a voting classifier of all the models tested but did not have time. Boosting was also on my list, hoping to get even better ensemble results.

Also, visualising the models, with graphs of the trees for example, would have been nice. For learning and explainability purposes but also for validating that the model does roughly what is expected.

Conclusions

- Ensemble methods are great 😄
- Explainability is something I value greatly