CPSC 330 - Applied Machine Learning

Tutorial 4

Course notes

So far, we have worked with various transformers and supervised machine learning models. The goal of this activity is to complete tables that provide an overview of

1. The strengths, weaknesses, and key hyperparameters of different machine learning models
2. The purpose, use cases, and key considerations of various transformers

(This will serve as a handy reference for your upcoming exam and beyond!)

Your task is to engage in group discussions and fill in the designated row in this Google document

- For strengths and weaknesses, some things to consider are:
  * concerns about underfitting
  * concerns about overfitting
  * speed
  * scalability for large data sets
  * interpretability
  * effectiveness on sparse data
  * ease of use for multi-class classification
  * ability to represent uncertainty
  * time/space complexity
  * etc.

Estimators

Solution:

Please note that this solution includes possible answers, but it is not comprehensive. More answers may have emerged during discussion with your peers and TAs

Model	Strengths	Weaknesses	Key hyperparameters (and their impact)
Decision trees	Robustness to unscaled data Somewhat interpretable (if small size)	Tendency to overfit/exhibit high variance Difficulty capturing interactions between	max_depth (direct correlation with complexity)

	Fast predictions Can rank features by importance Easy to extend to multi-class classification	variables	
KNN	No training time Potential to model complex decision boundaries Easy to extend to multi-class classification	Long prediction time Curse of dimensionality Unable to distinguish between strong/weak features Sensitive to unscaled data	k (inverse correlation with complexity)
SVM RBF	Very effective at modeling complex decision boundaries Perform well on high dimensional problems, even with small datasets	Struggle with large datasets Kernel may not be suitable for specific problem Need to be adapted for multi-class problems Sensitive to unscaled data	C and gamma (direct correlation with complexity)
Linear models (logistic regression or linear regression)	Interpretability Simplest, lightest weight model	Strong bias (simple models) Need to be adapted for multi-class problems Sensitive to unscaled data Performs best on linearly separable data	Alpha (inverse correlation with complexity) C (direct correlation with complexity)

Transformers

Transformation	Purpose	Use cases	Key consideration
Imputation	Replacing missing data with reasonable estimates	Most cases of datasets with missing data	Not all missing data are equal (could be missing not at random) More sophisticated imputation could be desirable
Scaling	Transform feature distribution to fit a specific scale/range	Required when estimators are sensitive to features having different ranges	Scaling choice may vary depending on initial distribution (e.g. presence of outliers) and application
One-hot encoding	Encoding categorical variables	Categorical (non ordered) variables	Must decide how to handle new labels in test set Too infrequent label values are not desirable, grouping may be advised
Ordinal encoding	Encoding ordinal variables	Categorical variables with a relative order between values	Encoded as integers but lacking numerical properties (e.g. can not add or subtract)
Bag-of-words encoding	Encoding text data	Encoding text data	Simplest option for encoding text data Several parameters to consider

Exercise 2 - Linear models

A professor is trying to create a linear regression model to predict exam scores for their students. The exam is scored out of 100 points. The predictors are hours studied (total), hours of sleep the night before the exam, and class attendance (encoded as low-medium-high).

The model is fitted on unscaled features, resulting in the following coefficients:

Feature	**Coefficient**
Hours studied	+4
Hours slept	+2
Attendance_low	-8
Attendance_medium	+2
Attendance_high	+8
Intercept	20

As the first step, write the linear model as a function below:

Predicted_score = *20 + 4(hours studied) + 2(hours slept) - 8(attendance low) + 2 (attendance medium) + 8 (attendance high)*

Then, answer the following questions:

1. Write a couple of scenarios where the model would predict a score of 100 for the student.

8 hours sleep, 14 hours study, high attendance
6 hours sleep, 19 hours study, low attendance

2. What is the meaning of the intercept?

The intercept corresponds to the model output when all inputs are = 0; it does not always make sense to interpret it

3. If a student was able to increase their attendance from low to medium, how much would their score go up (all other factors being equal)?

The score would go up by 10 points (difference between low and medium level)

4. Can you identify which feature has the greatest impact on the predicted score? Why or why not?

No, because the features have not been scaled.

5. A student has high attendance, studies 20 hours, and sleeps the recommended 8 hours before the exam: what is their predicted score? Does this reveal any problem with our model?

The output given these feature values would be 124, which is impossible given the maximum exam score of 100. This could mean that the model was fitted on different ranges for the features (e.g. less than 20 hours of study), or that the model is struggling with the non-linear nature of the problem (because the maximum score allowed is 100 and does not grow linearly beyond that point)