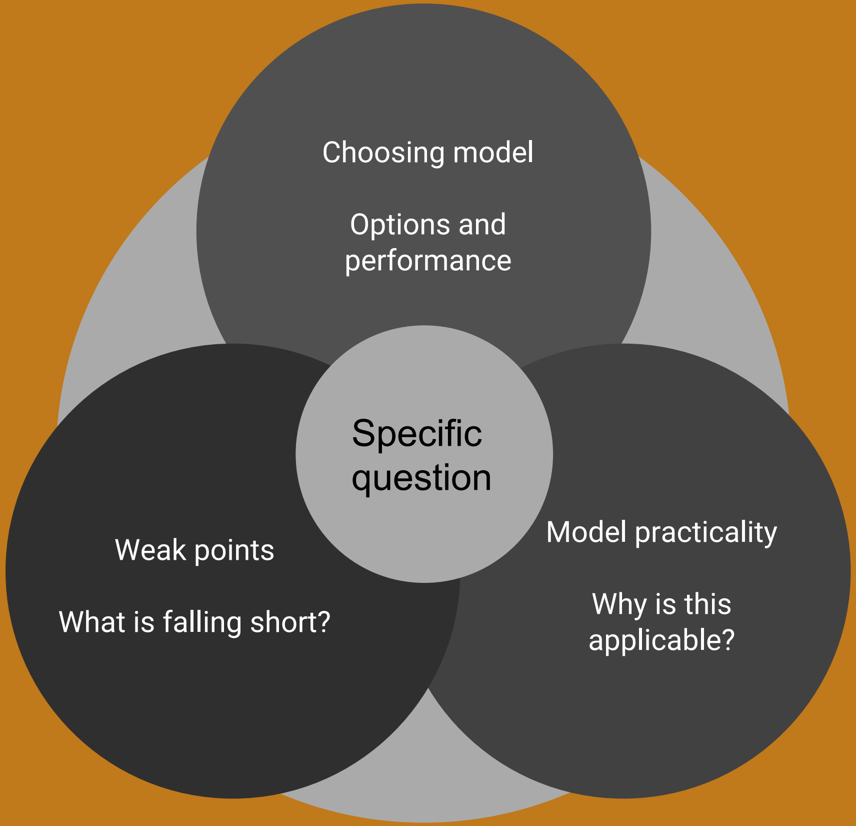
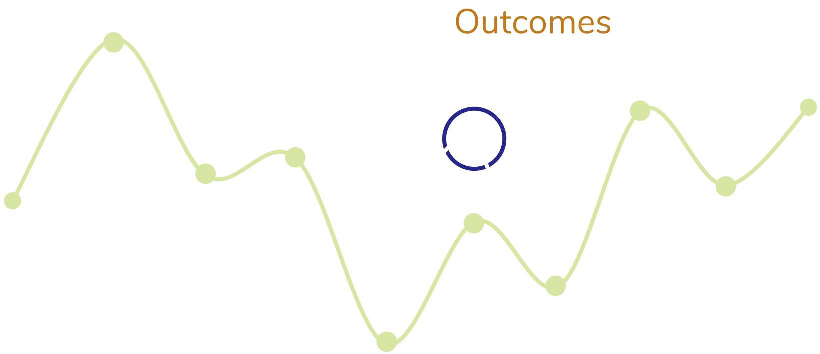
Supervised learning

An end-to-end project
https://tinyurl.com/y9u4tqm3


Choosing model
Options and performance
Specific question
Weak points
What is falling short?
Model practicality
Why is this applicable?

Adoption rates and outcomes



Research question: Are there features that can explain or predict the most likely outcome for animals at the Austin shelter?

Characteristics of the data

Context	Content	Approach
The data set comes from an open data initiative in Austin - the data was stored on servers on the Kaggle domain.	The dataset contains shelter outcomes of several types of animals and breeds from 10/1/2013 to the present with a hourly time frequency.	Outcome of interest: what can we predict? Animal status outcomes or adoption rates seemed like the best candidate.

High-level process: three steps

Consistency and quality | Initial models | Model tuning

Data preparation

The data quality is not model-ready. It needs transformations, selection, and filtering.

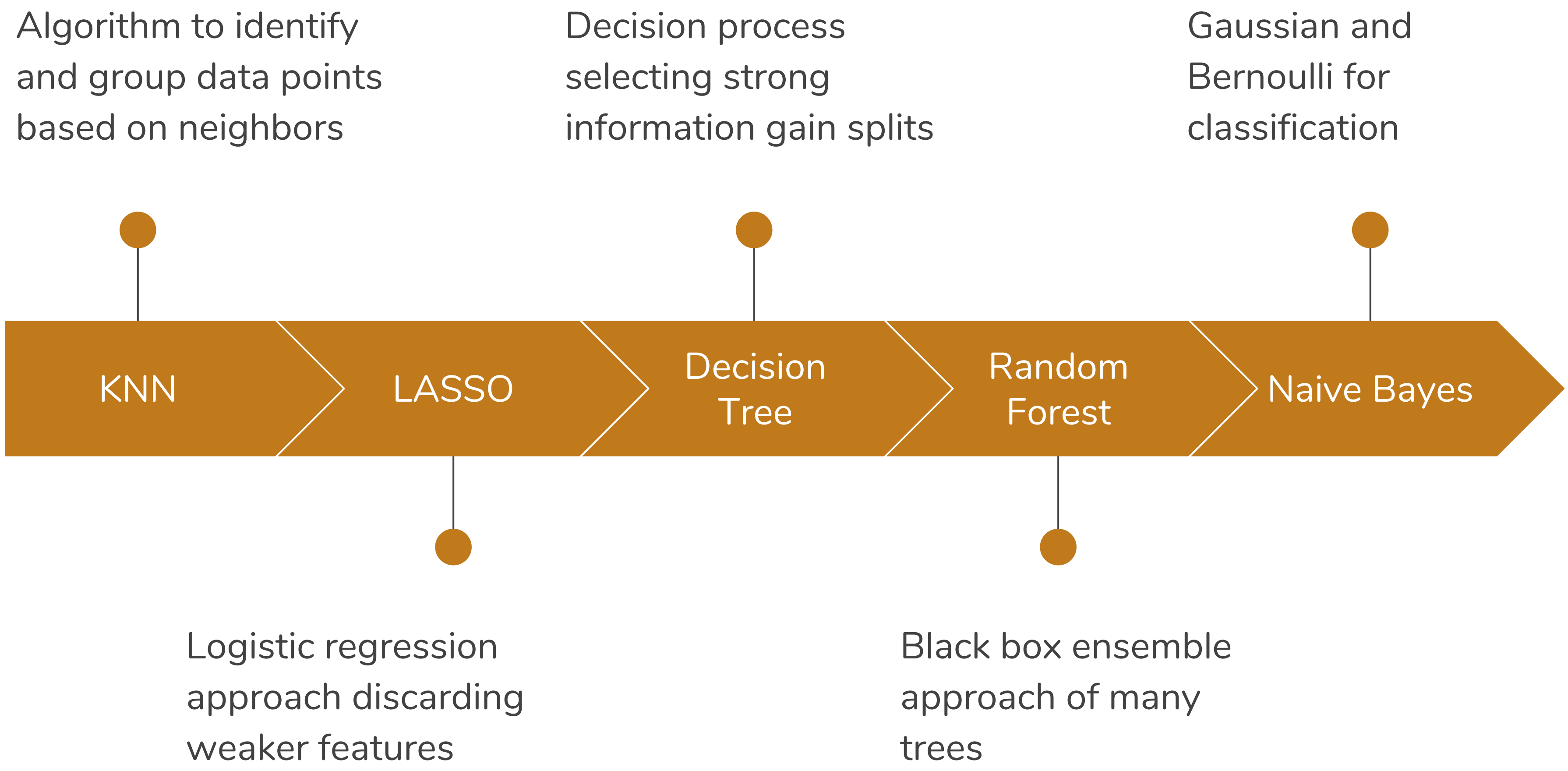
Classification models

The course has covered many different models for classification and regression. We have at least eight different models to attempt.

Increase performance

After finding inherently strong models against this data, we can tune a few of the best - single and ensemble.

Testing different models

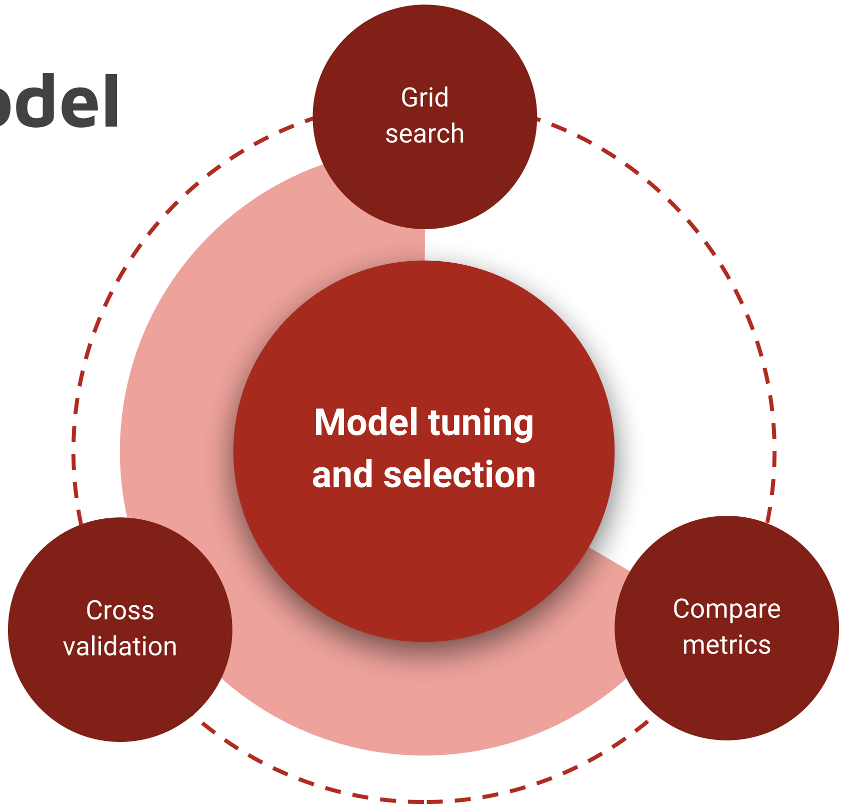

Algorithm to identify and group data points based on neighbors
Decision process selecting strong information gain splits
Gaussian and Bernoulli for classification
KNN
LASSO
Decision Tree
Random Forest
Naive Bayes
Logistic regression approach discarding weaker features
Black box ensemble approach of many trees

0.5-0.79

SKLearn metric scores across our models on binary outcomes

Improving the model

We started with a handful of classification algorithms. After checking the default parameters and their scores, we gather an idea of what is working. Then, after pushing the model through these steps, we can determine which are working, and if they might improve.



Fixing features: trade offs between retention or tossing

Binary outcomes from many options and some feature discarding may be ill-advised

Hitting more than 0.75 accuracy across a binary outcome is certainly an improvement over random guessing and chance. Our models were able to do this pretty early on - probably due to a healthy-sized, simple, and clean data set.

Revisiting the feature transformation and including some information previously left out was able to improve this at orders of magnitude. Hitting F1 scores of about 0.7 against random-guessing likelihoods of 0.25 is certainly trending in the right direction.

Models: best performers

Weaknesses: Overall, the numbers turned out to be decent. Hitting higher accuracies is always nice, but this is a trade off with potential overfitting. The inability of this data set to intuitively provide dozens of visualization options is probably the weakest part of its modeling process; its absence was noticed.

Pure predictive power: The Gradient boosting classifier seemed to pull some of the best scores off - and it does so against an F1 score. This covers several metrics to indicate performance. It *is* an ensemble method, though, and this does not tell us much about this question - *why*?

Explanatory power: K-Neighbors classifier does a solid job for a single model. The F1 score indicates it performs across different metrics. Luckily, this model does provide outcome insight - an explanation as to why.