MSDS600 W4: Tree-based ML and Feature Selection

REGIS UNIVERSITY

Review from W3

Types of ML:

- Supervised Learning
  - Classification (binary, multiclass, multilabel), regression
- Unsupervised
- Reinforcement
- Semi-supervised

Bias-variance tradeoff – high bias = underfitting, high variance = overfitting

Logistic regression is a relatively simple model for binary classification (works for multi-class classification too)

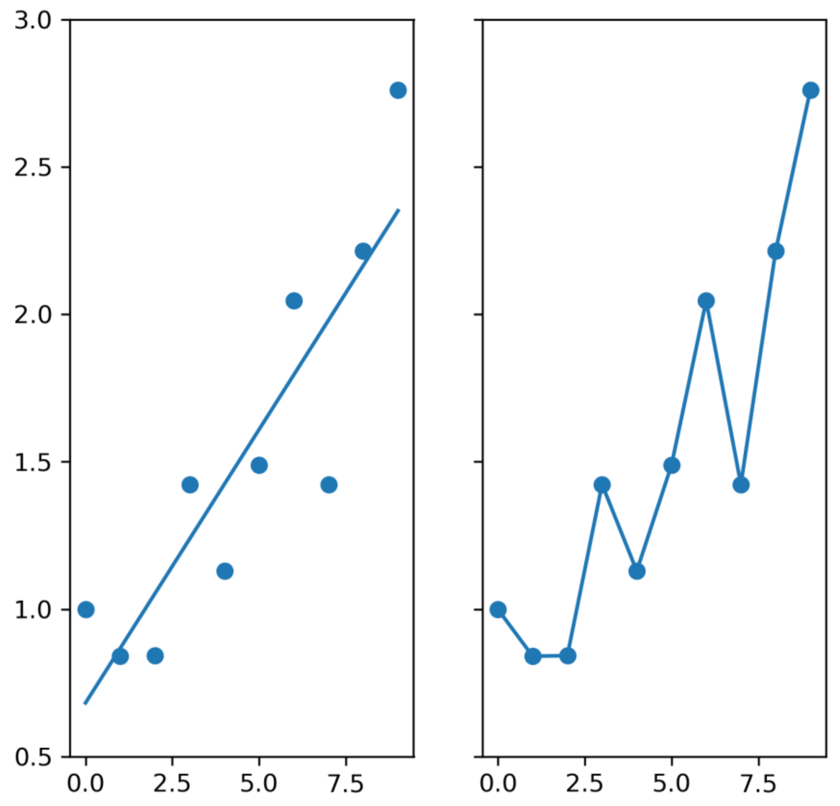
Train/test splits – evaluate performance on test set to check for overfitting

Linear fit

$y = mx + b$

Polynomial fit

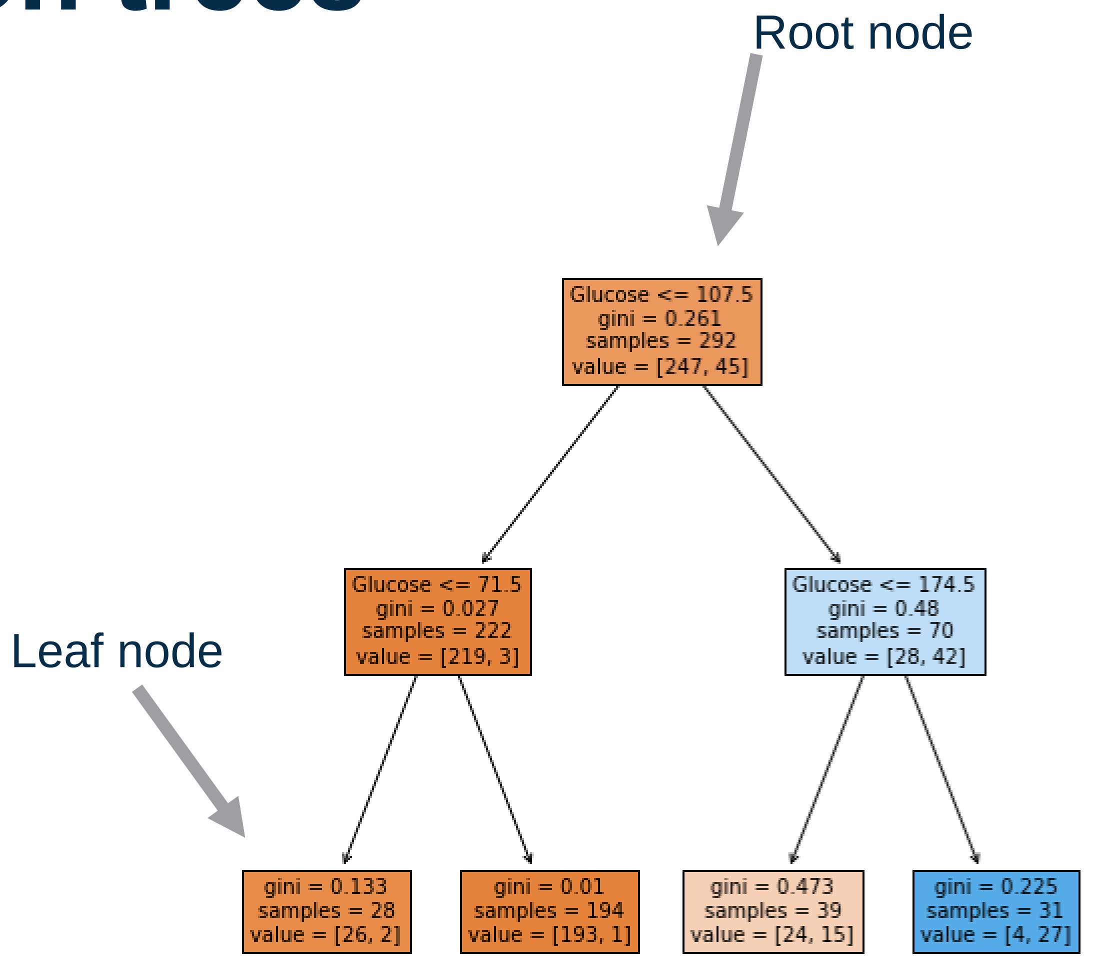
$y = m_1x + m_2x^2 + m_3x^3 + m_4x^4 + b$



REGIS UNIVERSITY

Decision trees

- Split data into groups automatically based on which values for which features give the best split between the classes.
- Optimize splits by trying different features and different splits and calculating purity of splitting categories (Gini impurity or entropy).
- Majority class in each leaf is the prediction.
- Can be used for regression too (average value of samples in leaves are the prediction).
- Data does not need to be scaled (unlike distance-based algos like KNN and other algos like neural networks)
- Prone to overfitting (high variance)

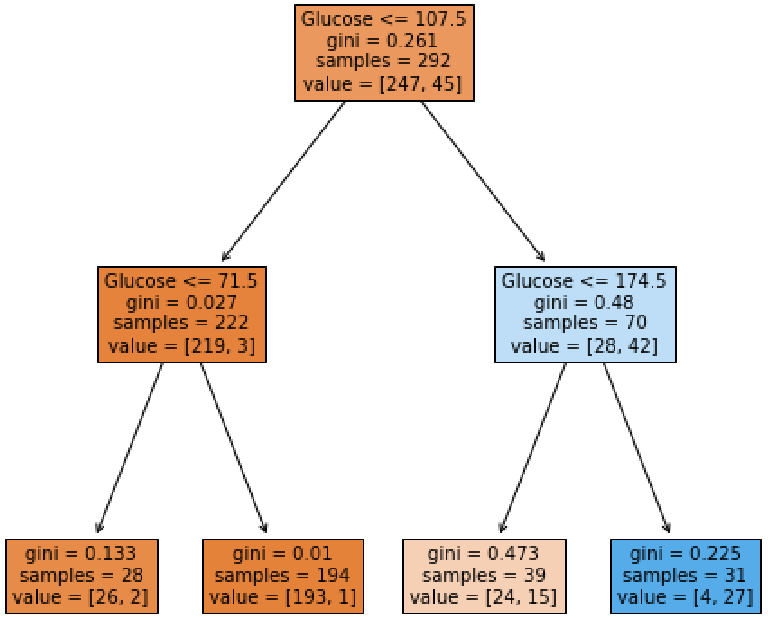


Gini criterion

Sometimes called Gini impurity or Gini index

In our example, the Gini value for the bottom left leaf is $1 - \left[\left(\frac{26}{28}\right)^2 + \left(\frac{2}{28}\right)^2\right] = 0.132$

For a perfect split (if it was 28/28 for class 0, or ‘no diabetes’) Gini is 0 (the minimum).



$$GiniIndex = 1 - \sum_j p_j^2$$

https://quantdare.com/decision-trees-gini-vs-entropy/

Entropy

- Similar to Gini, but slightly more complex
- Minimum (best) is 0, when the samples in a leaf node are all one class

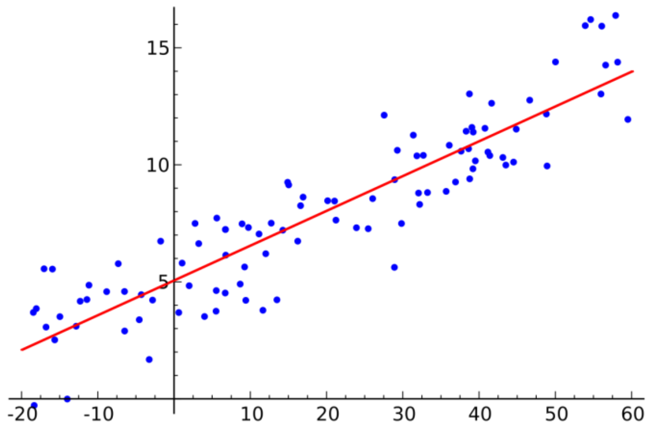
$$Entropy = -\sum_j p_j \cdot log_2 \cdot p_j$$

https://quantdare.com/decision-trees-gini-vs-entropy/

Regression

- Most ML algorithms in sklearn have a regression counterpart:
  - DecisionTreeClassifier has DecisionTreeRegressor, for example.
- These use a different criterion than what we've seen so far.
- Instead of Gini or Entropy, they can use things like MSE (mean squared error), which takes the difference between each prediction and actual value, squares it, then sums those up. There are other options available.
- The counterpart to logistic regression is linear regression.
- The sklearn documentation is a great resource, and has more information on decision trees here: https://scikit-learn.org/stable/modules/tree.html#tree

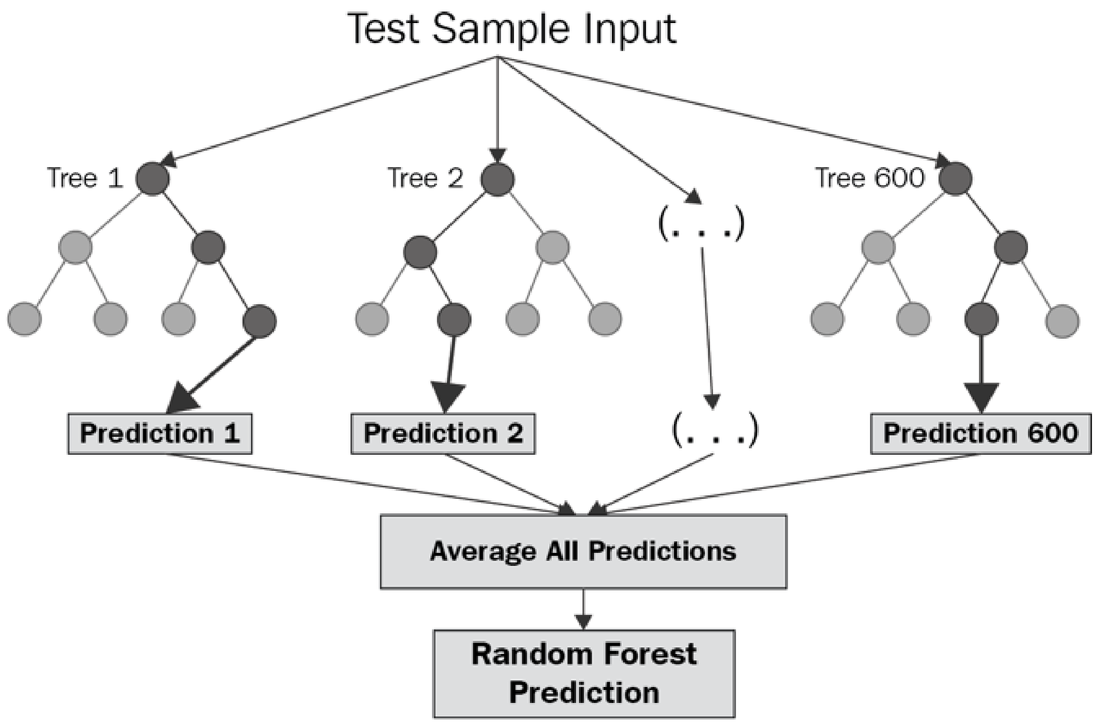
Linear regression image from Wikipedia:
https://en.wikipedia.org/wiki/Linear_regression



REGIS UNIVERSITY

Random forests

- Random forests take many decision trees together and average their predictions to get a final prediction.
- For each decision tree, we sample our data with replacement. So, we sample a data point from the entire dataset, and do this enough times so we have the same number of datapoints in our sample as in the original data. Some points will be in the sample multiple times, others won't be in there at all.
- For each split in each decision tree, we only try splitting on a random subset of features (the max_features hyperparameter in sklearn).
- We do this for a specified number of trees.
- There is both an classifier and regressor random forest class in sklearn.



https://corporatefinanceinstitute.com/resources/knowledge/other/random-forest/

Feature importances

The curse of dimensionality says that as we increase number of features, we need exponentially increasing numbers of datapoints to cover the sample space.

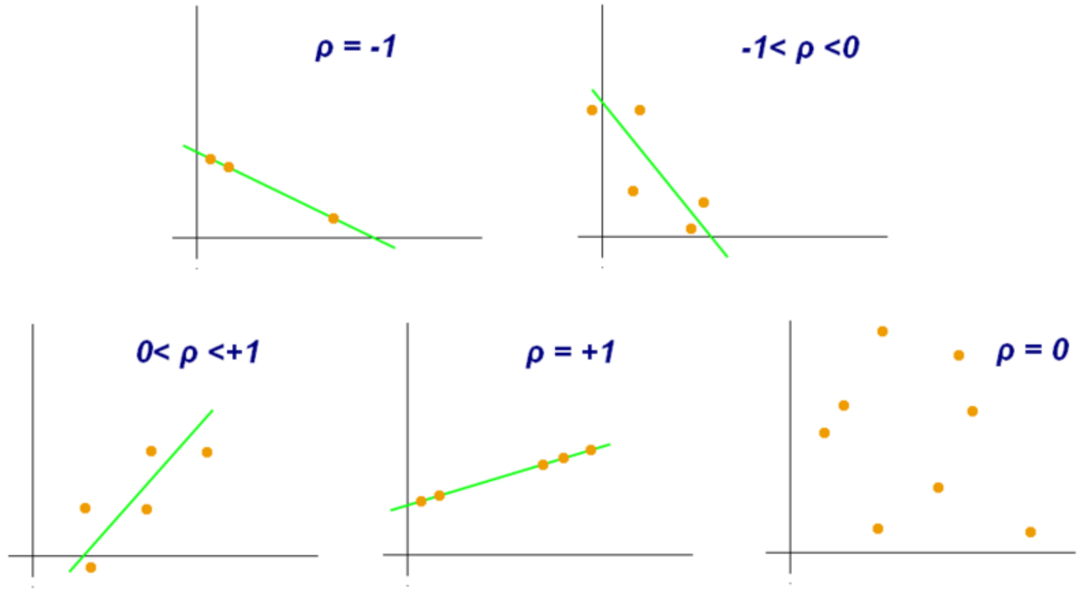
- We can either collect more data, or remove some features to combat this. If we have too many features, it can lead to poor performance of ML algorithms (causing over- or under-fitting depending on the algorithm).

We can select the best features in several ways:

- Univariate statistics (correlation, mutual information score, chi2 score, etc)
- Forward and backward selection with ML algos – add or subtract one feature at a time until we have a minimal number of features and good performance
- Recursive selection with ML algos – same idea as forward and backward selection, but removes one feature at a time (trying all features) until we reach a number of features or best performance
- Feature importance methods – rank features based on importance (e.g. Gini or Entropy gain for trees)

Pearson correlation

- Measures linear relationship strength between two variables
- -1 to +1, with 0 being no correlation and -1 and +1 being perfect correlation



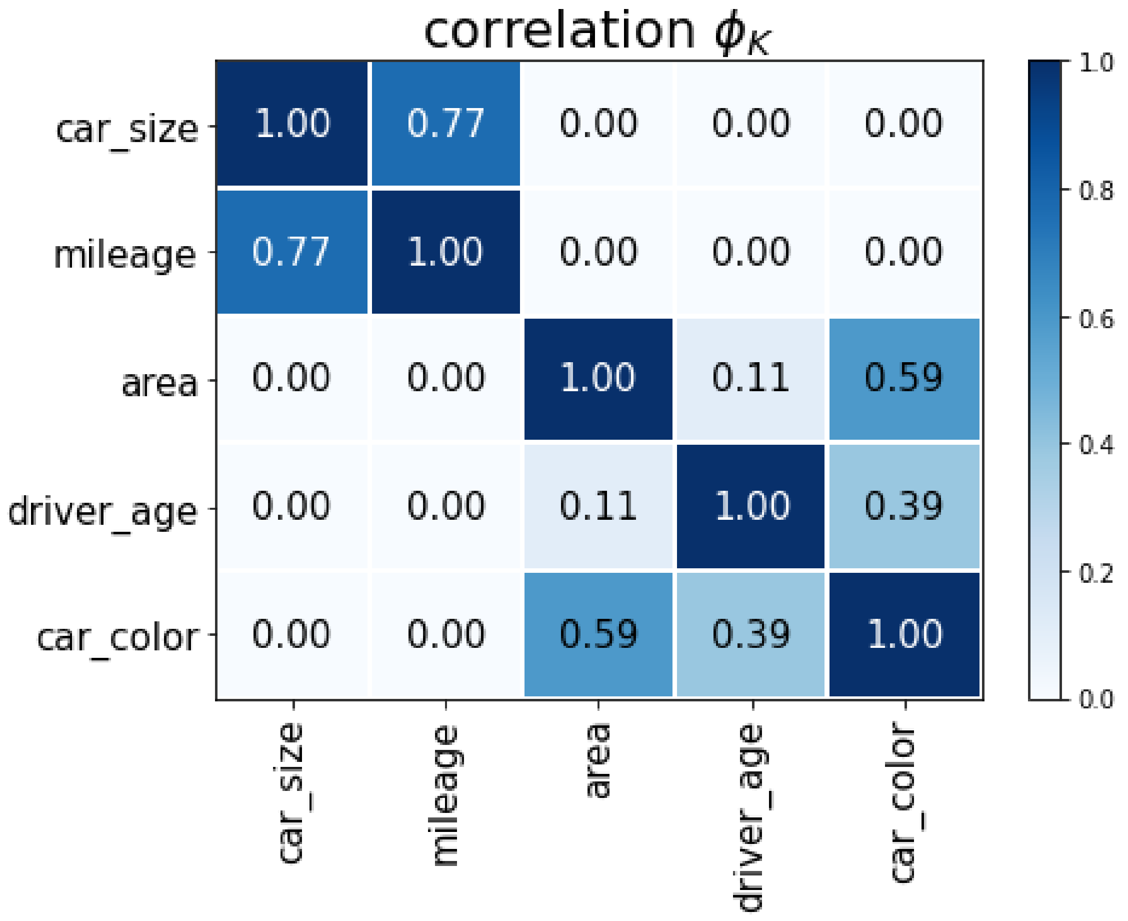
$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

https://en.wikipedia.org/wiki/Pearson_correlation_coefficient

REGIS UNIVERSITY

Phi-k correlation

- A newer version of correlation
- Bins numeric groups into categories and uses chi-2 correlation
- More info in the documentation and their published paper: https://phik.readthedocs.io/en/latest/



REGIS UNIVERSITY