Decision Tree notes

- A simpler model built from a large dataset will perform better than a complex model built with a small dataset.

Decision Tree

Decision trees represent rules, which can be understood by humans and used in knowledge systems such as a database

They are used for classification and prediction

The requirements for a decision tree are:

- Predictor Attribute: Attributes that measure some important feature of an object.
- Pre-defined target: The target variable is a discrete variable (Binary or Multi-class). Each object in a target variable has one of a set of mutually exclusive classes.
- Sufficient data: Enough training cases, whose target class is known, should be provided to build a model.

A binary target would only have two values that a result can take.

A multi-class discrete target would have more than two values that the resulting classification can take on.

A class refers to the value of an attribute. i.e For an attribute Humidity the classes could be {High, Normal, Low}.

Entropy

Entropy is the measure of uncertainty.

$H(X) = \sum_x P(x). \log \frac{1}{P(x)} = -\sum_x P(X) \log P(x) = -E[\log P(X)]$

The formula is made to satisfy certain principles

1. Uniform distributions have maximum uncertainty.
2. Uncertainty is additive for independent events
3. Adding an outcome with zero probability has no effect
4. The measure of uncertainty is continuous in all its arguments
5. Uniform distributions with more outcomes have more uncertainty
6. Events have non-negative uncertainty

Nice definition of a probability distribution: A function that assigns a probability to every possible outcome such that the probabilities add up to 1.

Two types of entropy need to be calculated to build a decision tree.

1. Entropy using the frequency table of one attribute: H(X)
2. Entropy using the frequency table of two attributes: H(X, Y)

Frequency table of two attributes

$H(Y, X) = -\sum_y \sum_x P(x, y) \log P(x, y)$

H(X, Y) = H(X) + H(Y|X)

Conditional Entropy

$H(Y|X) = -\sum_x \sum_y p(x, y) \log P(y|x)$

Mutual Information

Mutual information is a quantity that measures a relationship between two random variables that are sampled simultaneously. How much information is communicated, on average, in one random variable about another.

The mutual information between two random variables is zero if and only if the two variables are statistically independent.

$I(X, Y) = \sum_x \sum_y P(x, y) \log \frac{P(x,y)}{P(x)P(y)}$

P(X) and P(Y) are the marginal distributions of X and Y obtained through the marginalization process. Distribution of the random variable not considering the other.

$I(X, Y) = H(Y) - H(Y|X) \geq 0$

Information Gain

To quantify the information conveyed by each attribute a function is defined, I(s), which represents how much information is gained by knowing the predictor attribute such that

1. I(*) is a decreasing function of the probability p_i, with I(*) = 0 if $p_i = 1$
2. $I(s_i, s_j) = I(s_i) + I(s_j)$: Joint information gain is the sum of information gains.

Point 1 refers to the fact that you gain no information from knowing the predictor attribute of an event that is certain to occur.

Point 2 refers to the fact that since we have independent predictor variables, the amount of information gained by knowing about the outcome of the two variables is the sum of the two individual amounts of information.

$I(s_i) = \log \frac{1}{p_i} = -log(p_i)$

Try not to forget that a logarithm is the “The power one must raise a number (the base) to in order to get a number” ie $\log_2 16$ means what value do I raise 2 to in order to get that value 16 (it’s 4).

Overfitting

- Model is built to exactly match (perfectly predict) the dataset being used for training. Model reflects the noise in the dataset more than it does the relationship between independent and dependent variables.
- Would generate different responses for similar datasets.
- Problem is we don't actually know what is noise in the dataset and what isn't

Underfitting

- Model lacking sufficient training such that it doesn't understand anything about the relationships buried in the dataset and makes random, incoherent, predictions.
- Would generate different responses for similar datasets
- A model that fails to understand the underlying trends in the data.

Bias Variance Tradeoff

An Overfitted model would generate different responses for different for different datasets. This of this as *Variance*

An Underfitted model would generate similar responses for different datasets. Think of this as Bias. It indicated that the model does not understand an important pattern in the data.

Overfitting solutions

- Cross Validation
- Regularisation
- Ensemble Learning

Cross validation

A technique for model selection.

A good model is one which performs well on an unseen dataset i.e Something other than the training set

Workflow:

- Split the data into **k** subsets - Training
- Keep some data aside to validate the performance of the model - Validation

Example(4-fold cross validation)

1. Split the data into 4 subsets
2. Take one of the subsets as the testing dataset. The other 3 subsets are the training datasets
3. Train your model and validate against the testing subset

4. repeat steps one to three using a different subset as the testing set each time

At the end you'll end up with k different models. Generally you'll validate the models and choose the one that performs best.

The models can be different algorithms entirely or they could be could be the same algorithm with different parameter values.

Regularisation

Penalises models that are too complex e.g high number of branches in decision tree or highest degree of a polynomial in regression analysis

E(New Model) = E(Model) + αReg(Complexity)

A form of regression that shrinks the coefficient estimates towards zero. For a refresher on regression just go down below

Some common regularization algorithms are *adjusted R-square* and *Lasso regression*

Adjusted R-square Since R-square is always positive and always increases as the number of terms in a model increases, it can be misleading as to the correctness of your model.

Adjusted R-square Increases only if the new term improves the model more than would be expected by chance.

Decreases when a predictor improves the model than less than would be expected by chance

Can be negative

Lasso Regression A shrinkage model that attempts to shrink the data points towards a central point, like the mean.

Meant to encourage simple, sparse models.

Ensemble Learning

Ensemble learning is the creation of a model from many varying base models. The way in which the base models are combined can range from simple methods such as averaging or max voting to more complex ones such as Boosting or stacking.

Works because the overfitting component of various models cancel each other when combined

Various kinds of Ensemble learning could be

- Using different techniques for model creation

- Using different training datasets
- Using different IVs
- Using different model parameters

So what's the result of Ensemble learning?

- the most common outcome between all the different models (discrete outcome)
- The average of the outcomes of the different models (continuous outcome)

More advanced ways of combining the models are Bagging, Boosting and Stacking.

Random Forest

A Random Forest is the Ensembling of different decision trees.

The Trees could be developed from:

- different Training Sets (Bagging)
- Different IVs (Random Subspace Method)

Can give us an idea as to which IVs are more important.

(brief) Regression refresher

Technique used to describe a relationship between a dependent variable and one or more independent variables

Before going into regression I'll first define the correlation coefficient r

Correlation Coefficient (r): ranging from -1 to 1 it measures the correlation of data from strong correlation (1, -1) to weak correlation (0)

$$r = \frac{\sum xy}{\sqrt{\sum x^2}\sqrt{\sum y^2}}$$

where $x = x_i - \bar{x}$ and $y = y_i - \bar{y}$

Linear Regression: relation of an dependent variable to a single independent variable

$$\hat{y} = b_0 + b_1 x$$

where b_0 is a constant, b_1 is the regression coefficient, x is the value of the independent variable and $\hat{y}$ is the predicted value of the dependent variable

to find the the constant b_0 and the regression coefficient b_1 you use the following formulae

$$b_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2},\ b_0 = \bar{y} - b_1 \bar{x}$$

I should mention that this is called the least squares method

- Minimizes the squared difference between values

I should also mention that your x and y values will be coming from your training data

There are many forms of regression but this should give an idea of how regression functions

Additional Notes

R-Squared

$R^2 = 1 - \frac{Explained Variation}{Total Variation}$

A measure of how close the data is to the fitted regression line

Whereas correlation explains the strength of the relationship between an independent and dependent variable. R-Square explains to what extent variance in one variable explains variance in the second variable.

For example is the R^2 value of a model is 0.5 than approximately half of the observed variation can be explained by the model's inputs.

The higher the R-Square value the better the model fit (in general but this is not always the case)

Cannot assess whether the coefficient estimates and the predictions are biased –> Look at residual plots for this.

Residual plots

Can be used to access whether the observed error is consistent with the stochastic error.

stochastic is just a fancy way of saying random.

there are two parts to your model: deterministic and stochastic. Deterministic being the predictive part and the stochastic part being the rest. The radom error.

Looking at a residual plot the values should be symmetrically and randomly distributed with a mean of zero. There should be no predictive value in the plot. e.g I should not be able to say with confidence that x = 5 will produce a positive value.

If you get a non-random residual plot it is an indication that the deterministic part of your model is not capturing som explanatory information that is "leaking into the residuals" e.g a missing variable, higher-order term etc.