Date.....

Machine Learning

Evaluating performance for classification

1. **Accuracy** = $\frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$

↳ Accuracy useful when target classes are balanced

2. **Recall** = $\frac{\text{Number of True Positive}}{\text{Number of True Positive + Number of False Negative}}$

↳ Ability of a model to find all the relevant cases within a dataset

3. **Precision** = $\frac{\text{Number of True Positive}}{\text{Number of True Positive + Number of False Positive}}$

↳ Ability of a model to identify only the relevant point

4. **F1-score** = $2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$

↳ harmonic mean of precision and recall

↳ optimal blend of precision and recall

Date.....

Evaluating performance for Regression

↳ Evaluation metrics like accuracy or recall aren't useful for regression problems

↳ need metric designed for continuous values.

1. Mean Absolute Error (MAE)

↳ mean of the absolute value of errors

↳ easy to understand

$$\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Simply said it's the average of the difference between our prediction and true value.

↳ However, MAE won't punish large errors, so we have MSE.

2. Mean Squared Error (MSE)

↳ mean of the squared errors

↳ larger errors are noted more than with MAE (TO punish outliers)

↳ more popular

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

3. Root mean squared error (RMSE)

↳ root of MSE.

↳ so that the unit of metrics is same as Y

↳ most popular

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Date 12.11.2021

Machine Learning

1. Simple Linear Regression

Predicting a response using a single feature

↳ a method to predict dependent variable (Y) based on values of independent variable (X).

↳ it is assumed that the 2 variables are linearly related.

↳ hence, we try to find a linear function that predicts the response value(Y) as accurately as possible as a function of the feature of (X)

How to find best fit line

↳ we are trying to minimize the errors in prediction by finding `line of best fit'

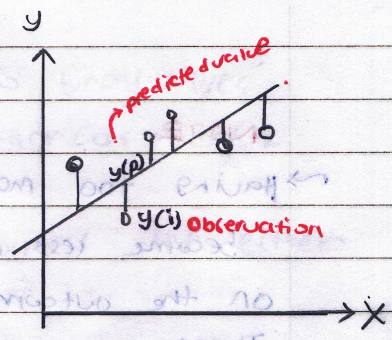
↳ trying to minimize the length between the observed value (Y_i) and the predicted value ~~(Yp)~~ from our model (Y_p)

$Y = b_0 + b_1 X_1$

- Dependent variable
- b_1: slope
- b_0: intercept
- X_1: independent variable

Y_i: observed value

Y_p: predicted value



~~min~~ $\min\{\text{sum}(y_i - y_p)^2\}$

2. Multiple Linear Regression

↳ similar to simple linear regression just that multiple linear regression attemps to model the relationship between 2 or more features

Dependent variable ← $Y = b_0 + b_1 X_1 + b_2 X_2 \ldots b_n X_n$ ↳ multiple independent variable

Date.....

For a successful regression analysis it is essential to validate these assumptions.

1. Linearity → The relationship btwn dependent and independent variables should be Linear.

2. Homoscedasticity → (constant variance) of the errors should be maintain

3. Multivariate Normality → Multiple regression assumes that the residuals are normally distributed

4. lack Of multicollinearity → it is assumed that there is little or no multicollinearity in the data.
Multicollinearity occurs when the features (or independent variables) are not independent of each other.

NOTE

→ Having too many variables could potentially cause our model to become less accurate. especially if certain variables have no effect on the outcome or have a significant effect on other variables. There are various methods to select the appropriate variable

1. Forward selection
2. Backward Elimination
3. Bi-Directional Comparison

Date 13.11.2021

3. Logistic Regression

- Used for classification problems
- to predict the group to which the current object under observation belongs to
- gives a discrete binary outcome between 0 to 1
- ex. to predict a person will vote or not in upcoming elections

How does it work?

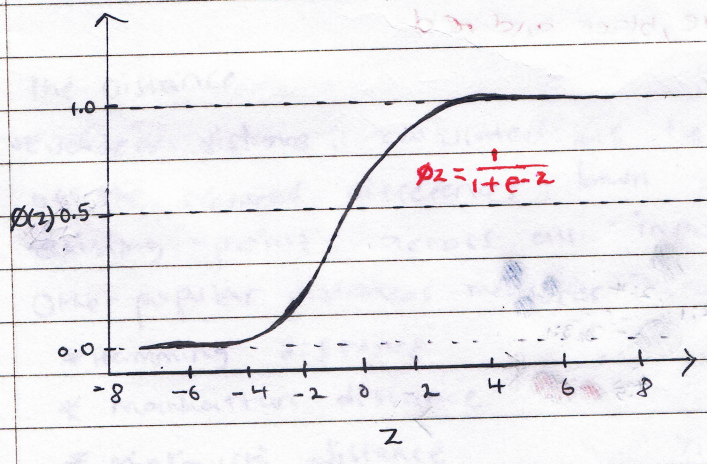
- Logistic Regression measures the relationship btwn the dependent variable (our label, what we want to predict) and the one or more independent variables (our features) by estimating probability using it's underlying logistic function.

Making predictions

- These probabilities must then be transformed into binary values in order to actually make a prediction. This is task of logistic function or sigmoid function
- This values btwn 0 to 1 and will be transformed into either 0 to 1 using threshold ~~functions~~ classifier.

Sigmoid function

- S-shaped curve that can take any real-valued number and map it into a value btwn range of 0 and 1, but never exactly at those limits.



Date.....

Type of Logistic Regression

1. Binary Logistic Regression
   → The categorical response has only 2 possible outcomes.
   ex: Spam or not

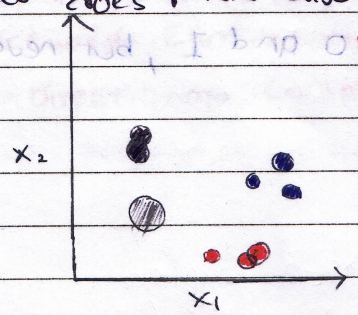
2. multinomial Logistic Regression
   → 3 or more categories without ordering.
   ex: predicting which food is preferred more (Veg, Non-Veg, Vega

3. ordinal logistic regression
   → 3 or more categories with ordering.
   ex: movie rating from 1 to 5

15.11.2021

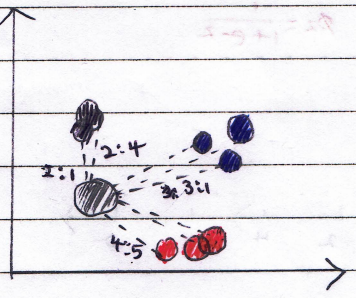
4. K Nearest Neighbours - KNN classification algorithm
   → simple yet most used classification algorithm
   → KNN is a non-parametric means that it does not make any assumptions on the underlying data distribution)
   instance-based (means that our algorithm doesn't explicit learn a model. Instead it chooses to memorize the training instances and used in a supervised learning setting

How does K-NN algorithm work?



we want to classify the grey point into 1 of 3 classes ~~right~~ blue, black and red

- Start by calculating the distance btwn the grey point and k-nearest points



Date.....

~~Making predictions~~

~~A) To~~

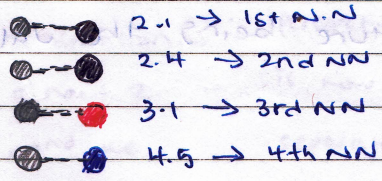
↳ KNN when used for classification - the output is a class membership (predicts a class - a discrete value).

↳ there are 3 key elements of this approach: a set of labelled objects, e.g. a set of stored records, a distance between objects, and the value of K, the number of nearest neighbors.

Making predictions.

↳ To classify an unlabelled object, the distance of this object to be labelled is computed, its k-nearest neighbors are identified, and the class label of the majority of nearest neighbors is then used to determine the class label of the object.

↳ for real valued input variables, the most popular distance measures is Euclidean distance.



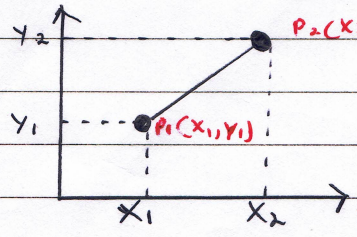
class	# of votes	
●	2	class ● wins the
●	1	vote
●	1	point ● is therefore predicted to be of class ●

The Distance

↳ Euclidean distance is calculated as the square root of the sum of the squared differences btwn a new point and an existing point across all input attributes.

Other popular distances measures include:

* Hamming distance
* Manhattan distance
* Minkowski distance



Euclidean Distance

$P_1 \text{ to } P_2 = \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$

Date.....

Value of K

- Finding value of K is not easy
- A small value of K means that noise will have a higher influence on the result and a large value make it computationally expensive.
- It depend a lot on your individual cases, sometimes it il best to ~~run~~ run through each possible value of K and decide for yourself.

18.11.2021

5. Support vector machines

What is svm?

- svm a supervised machine learning algorithm which can be used for both classification or regression, but mostly in classification problems.
- In this algorithm, we plot each data item as a point in n-dimensional space (where n is the number of features) with the value of each feature being the value of a particular coordinate.

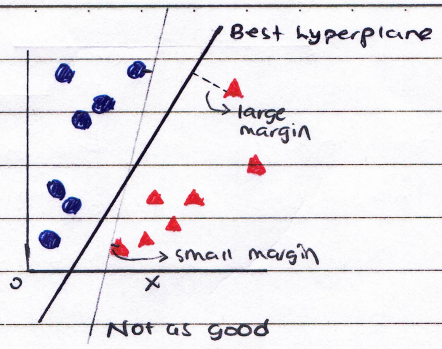
How is data classified?

- We perform classification by finding the hyperplane that differentiates the 2 ~~hyperplanes~~ classes very well.
- In other words the algorithm outputs an optimal hyperplane which categorizes new examples.

What is optimal Hyperplane?

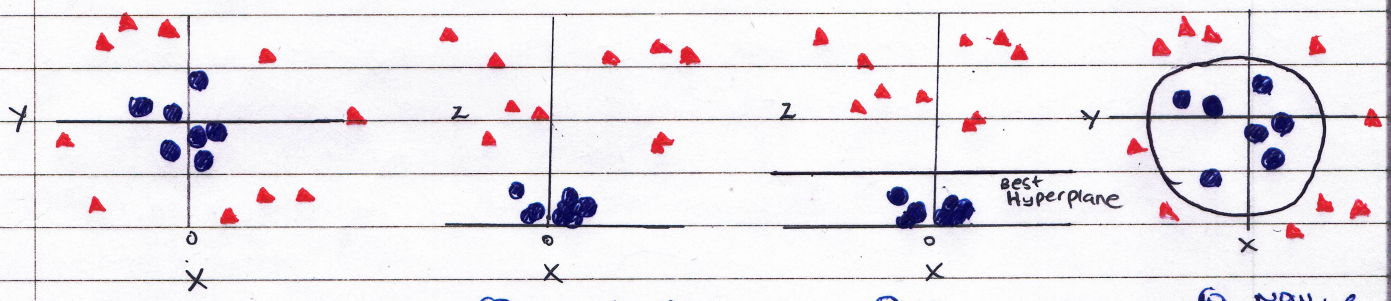
- For svm, it's the one that maximizes the margins from both tags.
- In other words: the hyperplane whose distance to the nearest element of each tag is the largest.

Date.....



* labelled data
* In 2D, the best hyper-plane is simply a line
* Not all hyper planes are created equal
* The best hyperplane is with a large margin

Nonlinear Data



① In above case we cannot draw a linear boundary. We will now add a 3rd dimension. We create a new Z dimension, and we rule that it can be calculated a certain way that is convenient for us: $z = x^2 + y^2$ (eqtn for a circle)

② This will give us a 3 dimensional space. From a diff perspective the data is now in linearly separated groups. All values for Z will be always positive because Z is squared sum of both X and Y

③ since we are in 3 dimensions now the hyper-plane is a plane parallel to the X-axis at a certain Z. We Choose the hyperplane, One that maximize the margins from both of the classes.

④ Now we map back to 2D. Our decision boundary is a circumference which sorts both tags using sum. We got a circle as hyper plane.

Tuning parameters

1. Kernel
2. Gamma
3. Regularization
4. margin.