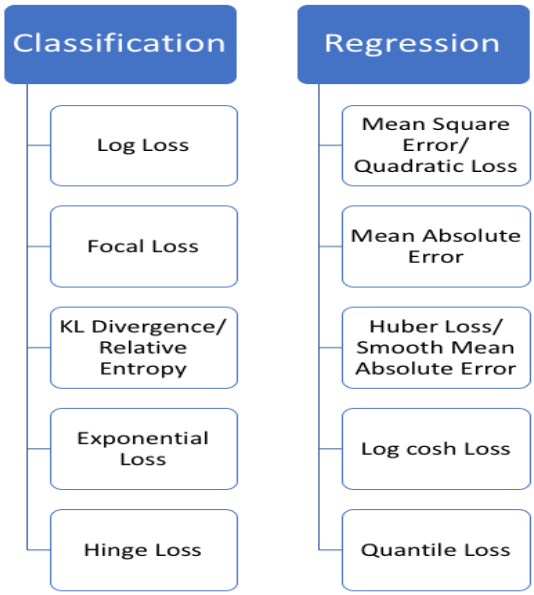
Survey on loss function, evaluation metrics and other components

Introduction

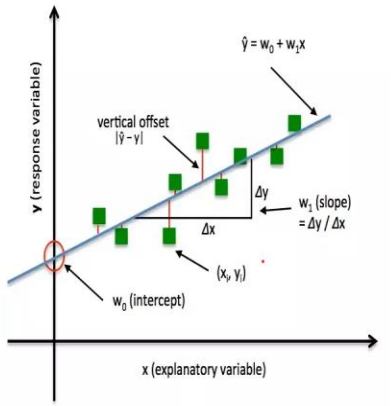
In deep learning, neural network help us to learn how to map a set of inputs to a set of outputs from the given training data. If you have given an image and wants to understand whether it is cat or dog, we will use convolutional filters. Because it is two dimensional and helps to extract different features step by step so that the information goes and turned to probability. It is trained using stochastic gradient descent and require to choose a loss function in the process of designing and configuring our model. In the learning process calculating the perfect weight to make good prediction is not easy. So to address this we can use backpropagation of error algorithm. Function which helps us to minimize or maximize set of weights is called the objective function or criterion. While minimizing criterion, we call it the cost function, loss function, or error function. In deep learning we aimed to minimize error that is why we concerned with loss function.

Loss Function

A loss function is a function which measure of how good a prediction model does in terms of being able to predict the expected outcome. For instance, in text recognition its basic rule is collapse repeated character which are not separated by blanks. This functions are in general categorized in to two: Classification and Regression Loss. Here is diagram which summarize types of loss function [1] and gradient plot [2].



(Types of loss function diagram)



(Loss gradient plot)

Let's stack on most popular loss functions and see how they defined.

a) Mean squared error
It is calculated by taking the difference between your predictions and the ground truth, square it, and average it out across the whole dataset. Which is implemented using the code:

```
def MSE(y_predicted, y):
    squared_error = (y_predicted - y) ** 2
    sum_squared_error = np.sum(squared_error)
    mse = sum_squared_error / y.size
    return(mse)
```

b) Log loss (cross entropy loss): Is given by formula $-(y\log(p)+(1-y)\log(1-p))$

Evaluation metrics

One goal of building machine learning model is to gain constructive feedback. That is it is not only building predictive models, it is about creating and selecting a model which gives high accuracy on out of sample data. Choosing the best evaluation metrics depend on type and the implementation plan of your model. Below we will see some of the evaluation metrics which is used to evaluate accuracy of our model. For instance, if our predictive model has probabilistic output (eg. random forest, logistic regression etc.) we can use mean square error, RMSE and the like. If the model is classification (eg. KNN) we can use F1 score, confusion metrics[3]:

- **Accuracy** : the proportion of the total number of predictions that were correct.
- **Positive Predictive Value or Precision** : the proportion of positive cases that were correctly identified.
- **Negative Predictive Value** : the proportion of negative cases that were correctly identified.
- **Sensitivity or Recall** : the proportion of actual positive cases which are correctly identified.
- **Specificity** : the proportion of actual negative cases which are correctly identified.

Confusion Matrix		Target			
		Positive	Negative		
Model	Positive	a	b	*Positive Predictive Value*	a/(a+b)
	Negative	c	d	*Negative Predictive Value*	d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)	
		a/(a+c)	d/(b+d)		

Example: In our case speech to text recognition, we will use CTC loss which is computed as the probability of the network predicting the correct sequence. It has a ground truth target transcript and tries to train the network to maximize the probability of outputting that correct transcript. Our metric is Word Error Rate (WER) that compares the predicted output and the target transcript, word by word (or character by character)

$$\text{Word Error Rate} = \frac{\text{Inserted} + \text{Deleted} + \text{Substituted}}{\text{Total words in transcript}}$$

Reference

1. https://heartbeat.fritz.ai/5-regression-loss-functions-all-machine-learners-should-know-4fb140e9d4b0
2. https://algorithmia.com/blog/introduction-to-loss-functions
3. https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/