PyData : Conference Mission Statement

- PyData is a gathering of users and developers of data analysis tools in Python.
- The goals are to provide Python enthusiasts a place to share ideas and learn from each other about how best to apply our language and tools to ever-evolving challenges in the vast realm of data management, processing, analytics, and visualization.

PyData : Conference Mission Statement

- We aim to be an accessible, community-driven conference, with tutorials for novices, advanced topical workshops for practitioners, and opportunities for package developers and users to meet in person.
- A major goal of the conference is to provide a venue for users across all the various domains of data analysis to share their experiences and their techniques, as well as highlight the triumphs and potential pitfalls of using Python for certain kinds of problems.

- A predictive model is not magic and it is not rocket science. It is actually a mathematical representation of reality that offers a view based on science and statistics that uses data that is currently available to predict the risk of an adverse event (or a preferred event) at some point in the future.
- The models, and there are many, look for patterns in the data that may not always be evident and which seem linked to an outcome of some sort.

- Depending on the data and the model it can do so in either a supervised learning fashion, where the model describes a relationship between a set of independent attributes and a dependent attribute or in an unsupervised learning fashion where the model, itself, will find relationships in the data without reference to independent or dependent variables.

Step 1 : Defining the Problem

Step 1 is defining the problem. Put simply, what question(s) are you trying to answer? Once you understand that you need to then think about what data is available to you to answer the question:

- Is the data directly related to the question?
- If it is not, can you create a proxy relationship to be able to link it?
- Is the data you need even available within the enterprise or elsewhere?

Step 1 : Defining the Problem

As part of step 1, you also need to specify the inputs and outputs of the model you are going to build as these may change as you change and tweak the model. Finally, dont forget the most commonly forgotten piece of any new initiative. Determine, up front, how you are going to measure the results.

- What measure of accuracy are you going to use? Is that level of accuracy good enough for the business?
- How will you benchmark the results?
- What criteria are you going to use to determine success or failure?

Step 2 : Processing the Data

- Step 2 is more rote and technical process the data. Collect the data (more is always better in this analysts mind but more does not always mean easier or better results).
- In general, more recent data is better and the data need to be consistent.
- Dont skimp on cleaning the data. While this may end up taking the most time, it is critical and erroneous data will create erroneous results.

Step 2 : Processing the Data

- Transforming the data is also worth the time and effort to improve the modeling process including such things as:
  - Converting non-numerical data to numeric (or vice versa)
  - Standardizing thing such as coding, definitions, costs, combining variables, etc.
- Predixion gives you tools, built into the software, to help you analyze, clean, and classify your data on the front end.

Step 3 : Run the Initial Model

- Step 3 is running the initial model. Part of step 3 is to split the dataset into a test dataset and a validation dataset. If you really want to be able to test the accuracy of the model once it has been built, you need to do this.
- The software will walk you through this and will do this for you by default, holding back 30% of the total data for validation testing, although it does allow you to override the default values.
- This is also the step whereby you will choose the method or methods by which you want to build the model and process the data.

Step 3 : Run the Initial Model

- As you become more familiar with predictive modeling and with your own data you will find that certain types of data and certain types of analyses or problems do lend themselves more or less to certain types of modeling.
- But if you are just starting out, you can use the software to guide you in the choice of a model or simply choose to run all the models against your data. Once done, run the model and move on to step 4.

Step 4 : Evaluate the Initial Model

- Step 4 is to evaluate the initial results. Are the results acceptable? Are they what you were expecting to see? Do you understand the results?
- Do they answer the question you are trying to answer?
- If the answer is yes, then move on to the next step.

Step 4 : Evaluate the Initial Model

If the answer is no, then consider the following:

- Try using different algorithms/models
- Try using different data elements or repackaging what you have
- Consider collecting more or different data
- Consider redefining the problem, changing the question and the means to an answer as you better understand your data and your environment

- Part of the learning process may be to try and not boil the ocean with your models.
- Think about setting up the model to run on a number of scenarios starting from the simple and most straightforward and then progressing to more and more complex.

Step 5

Step 5 is to select the final model.

- Dont be afraid to try a number of different models and then when you are satisfied with the results, choose the best one.
- We will talk about means of assessing the accuracy of your model in a bit.
- For now, choose the final model and consider whether you want to rerun the entire dataset against the selected model and re-examine the results

Step 6 : Testing the Model

Step 6 is to test the final model. This is another one of those things that often seems not to get done. It is important to test the final model and the only way to do so is to take the selected model and run it against a second, unrelated dataset (e.g. - the validation dataset or the portion of the dataset that was held back for this purpose) and assess the results.

Step 6

- Do not tweak or change the model in any way at this point as it will invalidate any comparison to the initial results.
- If the results are similar and you are satisfied with them you can move on to the final step.
- If you are not, then go back (to step 3) to reassessing the model and the data, make any necessary or desired changes and try re-running the model again.

Step 7 : Use the Model

Step 7 is to apply the model and run the prediction. There are actually two parts to this. One is done if you want to refine the model then you can use the output from the model to determine next steps and potential intervention or changes. Continue to test the model as much or as often as needed. But when you are satisfied, please do two things:

1. Run the necessary measures to test the final accuracy of the model (see our next discussion points)
2. Take the output from the model and turn it back into language and output for the business that answers the initial question you set out to answer and makes it useful/usable for them

SUPERVISED LEARNING

Applications in which the training data comprises examples of the input vectors along with their corresponding target vectors are known as supervised learning problems.

Supervised learning is when the data you feed your algorithm is "tagged" to help your logic make decisions.

Example: Bayes spam filtering, where you have to flag an item as spam to refine the results.

UNSUPERVISED LEARNING

In other pattern recognition problems, the training data consists of a set of input vectors x without any corresponding target values. The goal in such unsupervised learning problems may be to discover groups of similar examples within the data, where it is called clustering

Pattern Recognition and Machine Learning (Bishop, 2006)

Unsupervised learning are types of algorithms that try to find correlations without any external inputs other than the raw data. Example: datamining clustering algorithms.

Types of Predictive Analytics Problems

- What is Classification and Regression
- Binary Classification Problems
- Logistic Regression

Confusion Matrix

The Confusion Matrix

The confusion matrix indicates that there were four true negative predictions, three true positive predictions, two false negative predictions, and one false positive prediction. Confusion matrices become more useful in multi-class problems, in which it can be difficult to determine the most frequent types of errors.

Accuracy

Accuracy measures a fraction of the classifier's predictions that are correct.

Accuracy

However, accuracy is not an informative metric if the proportions of the classes are skewed in the population. For example, a classifier that predicts whether or not credit card transactions are fraudulent may be more sensitive to false negatives than to false positives. To promote customer satisfaction, the credit card company may prefer to risk verifying legitimate transactions than risk ignoring a fraudulent transaction.

Accuracy

Because most transactions are legitimate, accuracy is not an appropriate metric for this problem. A classifier that always predicts that transactions are legitimate could have a high accuracy score, but would not be useful. For these reasons, classifiers are often evaluated using two additional measures called precision and recall.

Precision and Recall

Recall from Chapter 1, The Fundamentals of Machine Learning, that precision is the fraction of positive predictions that are correct. For instance, in our SMS spam classifier, precision is the fraction of messages classified as spam that are actually spam. Precision is given by the following ratio:

$$P = \frac{TP}{TP + FP}$$

Recall

- called sensitivity in medical domains, recall is the fraction of the truly positive instances that the classifier recognizes. A recall score of one indicates that the classifier did not make any false negative predictions.
- For our SMS spam classifier, recall is the fraction of spam messages that were truly classified as spam. Recall is calculated with the following ratio:

$$\text{Recall} = \frac{tp}{tp + fn}$$

Precision and Recall

- Individually, precision and recall are seldom informative; they are both incomplete views of a classifier's performance.
- Both precision and recall can fail to distinguish classifiers that perform well from certain types of classifiers that perform poorly.
- A trivial classifier could easily achieve a perfect recall score by predicting positive for every instance.

Precision and Recall

- For example, assume that a test set contains ten positive examples and ten negative examples.
- A classifier that predicts positive for every example will achieve a recall of one, as follows:

10 1 10 0 R = = +

- A classifier that predicts negative for every example, or that makes only false positive and true negative predictions, will achieve a recall score of zero.
- Similarly, a classifier that predicts that only a single instance is positive and happens to be correct will achieve perfect precision.

Cross Validation

The confusion table is a table in which the rows are the observed categories of the dependent and the columns are the predicted categories. When prediction is perfect all cases will lie on the diagonal. The percentage of cases on the diagonal is the percentage of correct classifications.

Cross Validation

The cross validated set of data is a more honest presentation of the power of the discriminant function than that provided by the original classifications and often produces a poorer outcome. The cross validation is often termed a jack-knife classification, in that it successively classifies **all cases but one** to develop a discriminant function and then categorizes the case that was left out. This process is repeated with each case left out in turn.This is known as leave-1-out cross validation.

This cross validation produces a more reliable function. The argument behind it is that one should not use the case you are trying to predict as part of the categorization process.

Error Rates

- We can evaluate error rates by means of a training sample (to construct the discrimination rule) and a test sample.
- An optimistic error rate is obtained by reclassifying the training data. (In the **training data** sets, how many cases were misclassified). This is known as the **apparent error rate**.
- The apparent error rate is obtained by using in the training set to estimate the error rates.
- It can be severely optimistically biased, particularly for complex classifiers, and in the presence of over-fitted models.

- If an independent test sample is used for classifying, we arrive at the **true error rate**.
- The true error rate (or conditional error rate) of a classifier is the expected probability of misclassifying a randomly selected pattern.
- It is the error rate of an infinitely large test set drawn from the same distribution as the training data.

Misclassification Cost

- As in all statistical procedures it is helpful to use diagnostic procedures to asses the efficacy of the discriminant analysis. We use **cross-validation** to assess the classification probability. Typically you are going to have some prior rule as to what is an **acceptable misclassification rate**.
- Those rules might involve things like, "what is the cost of misclassification?" Consider a medical study where you might be able to diagnose cancer.
- There are really two alternative costs. The cost of misclassifying someone as having cancer when they don't. This could cause a certain amount of emotional grief. Additionally there would be the substantial cost of unnecessary treatment.

There is also the alternative cost of misclassifying someone as not having cancer when in fact they do have it.
A good classification procedure should

- result in few misclassifications
- take **prior probabilities of occurrence** into account
- consider the cost of misclassification

For example, suppose there tend to be more financially sound firms than bankrupt firm. If we really believe that the prior probability of a financially distressed and ultimately bankrupted firm is very small, then one should classify a randomly selected firm as non-bankrupt unless the data overwhelmingly favor bankruptcy.

- There are two costs associated with discriminant analysis classification: The true misclassification cost per class, and the expected misclassification cost (ECM) per observation.
- Suppose there we have a binary classification system, with two classes: class 1 and class 2.
- Suppose that classifying a class 1 object as belonging to class 2 represents a more serious error than classifying a class 2 object as belonging to class 1.
- There would an assignable cost to each error. $c(i|j)$ is the cost of classifying an observation into class j if its true class is i.

Misclassification

The costs of misclassification can be defined by a cost matrix.

	Predicted Class 1	Predicted Class 2
Class 1	0	$c(2\|1)$
Class 2	$c(1\|2)$	0

Expected cost of misclassification (ECM)

- Let p_1 and p_2 be the prior probability of class 1 and class 2 respectively. Necessarily $p_1 + p_2 = 1$.
- The conditional probability of classifying an object as class 1 when it is in fact from class 2 is denoted $p(1|2)$.
- Similarly the conditional probability of classifying an object as class 2 when it is in fact from class 1 is denoted $p(2|1)$.

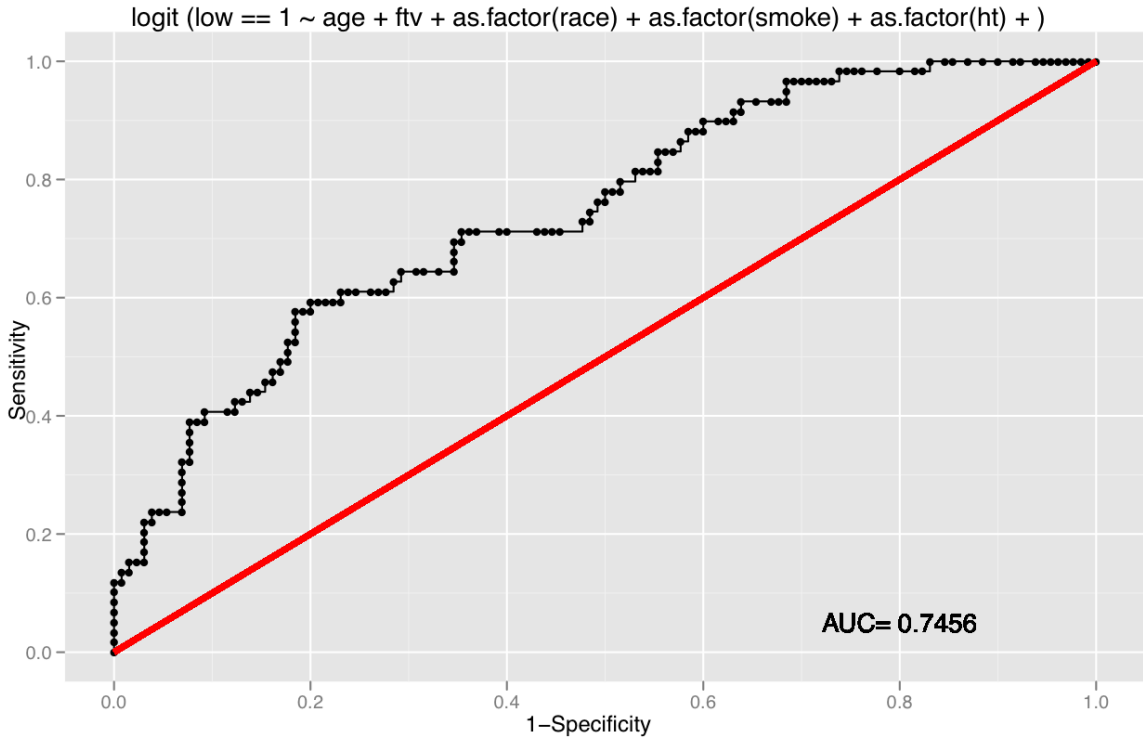
$$ECM = c(2|1)p(2|1)p_1 + c(1|2)p(1|2)p_2$$

(In other words: the sum of the cost of misclassification times the (joint) probability of that misclassification.

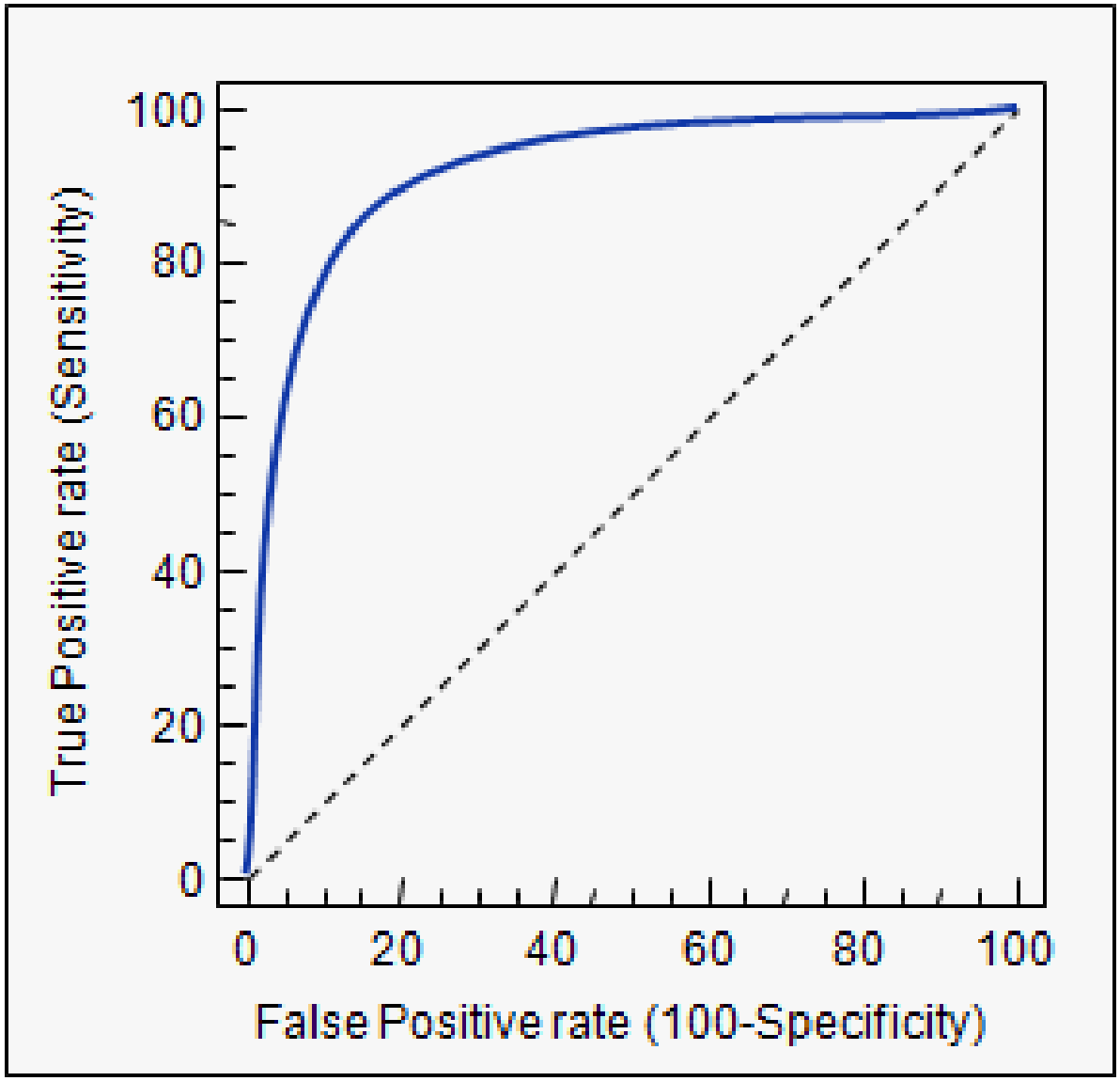
- A reasonable classification rule should have ECM as small as possible.

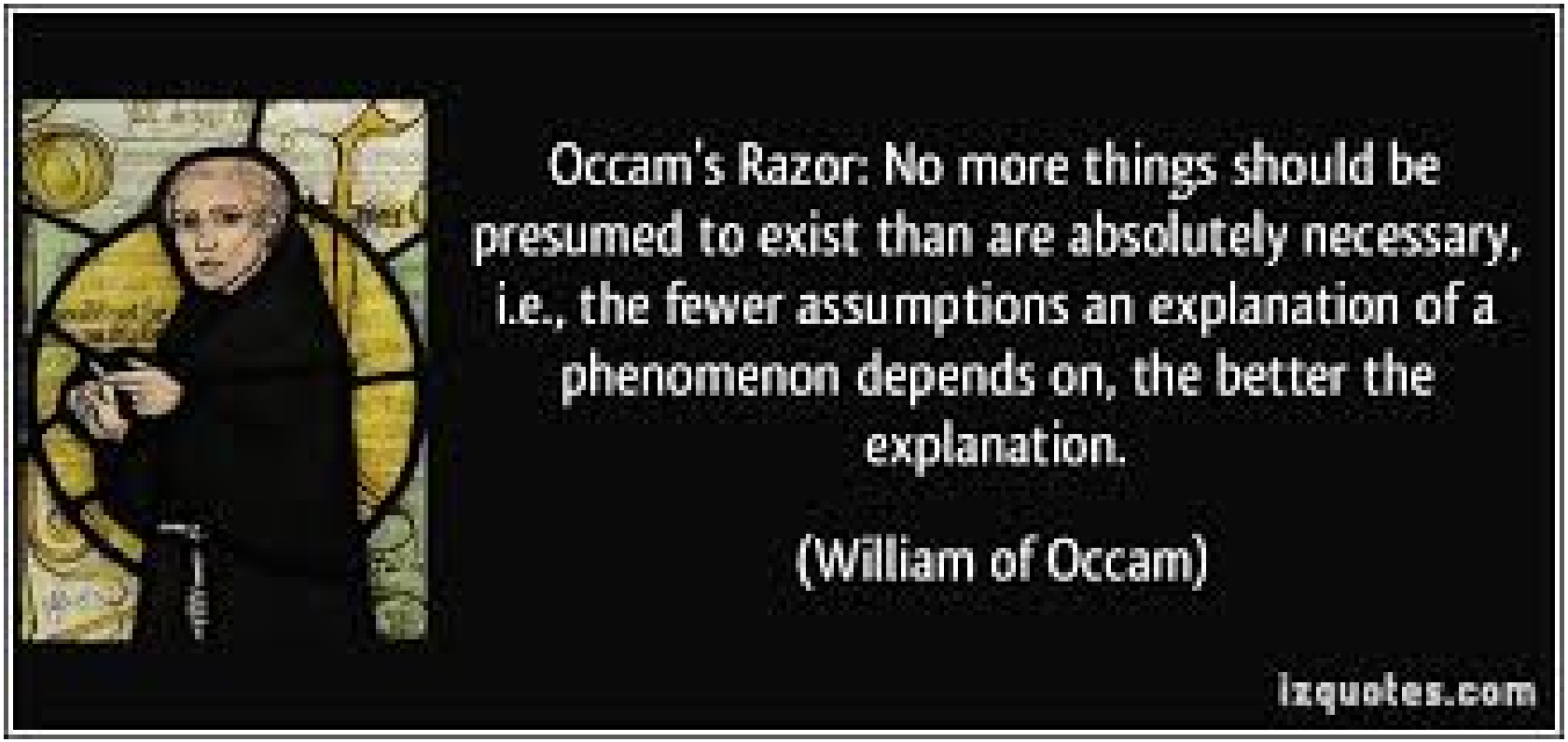
ROC curves

-
-


logit (low == 1 ~ age + ftv + as.factor(race) + as.factor(smoke) + as.factor(ht) +)
Sensitivity
1−Specificity
0.0
0.2
0.4
0.6
0.8
1.0
AUC= 0.7456

AUC or Area Under the Curve is a single numerical measure of predictive power.




Occam's Razor: No more things should be presumed to exist than are absolutely necessary, i.e., the fewer assumptions an explanation of a phenomenon depends on, the better the explanation.
(William of Occam)
lzquotes.com

- Occam's razor (or Ockham's razor) is a principle from philosophy. Suppose there exist two explanations for an occurrence.
- In this case the simpler one is usually better.
- Another way of saying it is that the more assumptions you have to make, the more unlikely an explanation is.
- Occam's razor applies especially in the philosophy of science, but also more generally.