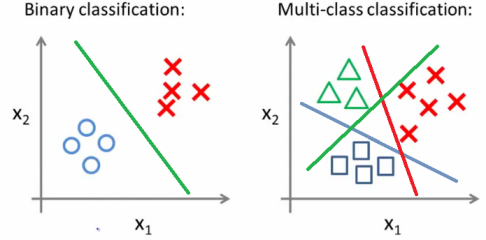
**The Problem: Multiclass classification:**

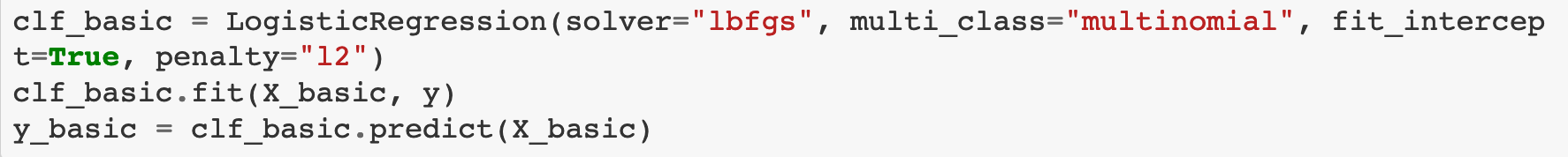
* The problem of classifying instances into one of three or more classes
* A common issue with this type of problem is that the data could be unbalanced to one class
  + In our case, the trouble is that since there are a few crops with a lot of data, the model that we build could be biased towards those crops
* Even if the data is balanced, most classifiers are limited to distinguishing between a handful of classes
  + We can train binary models as multiclass classifiers using one-vs-rest and one-vs-one, or multinomial classification support in logistic regression



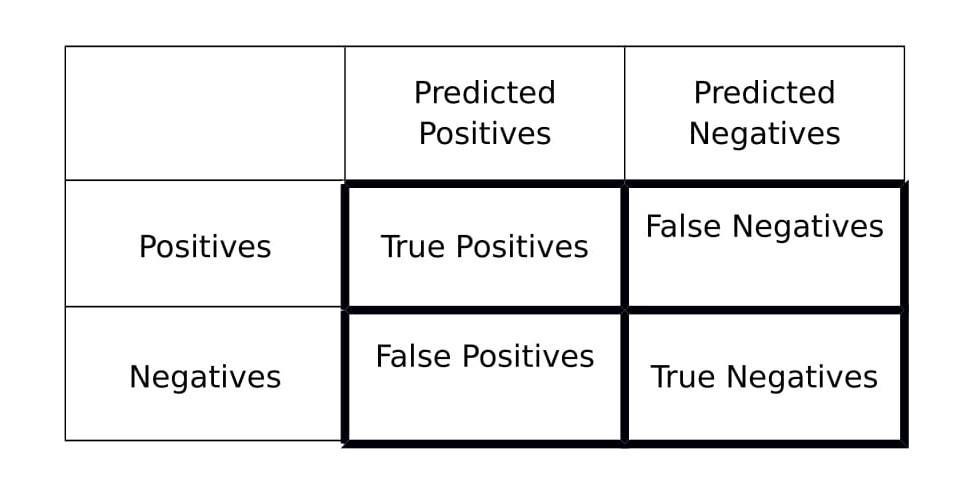
**The Pipeline:**

* Fixing the imbalanced data
  + Since we are working with a relatively small, imbalanced dataset, we’d want to over-sample our data or provide bias to minority classes
    - We can oversample using SMOTE using the imblearn package
    - We can add bias by using class\_weight from sklearn
  + Normalize data
  + One-hot-encode categorical data?
  + Train/test split the data

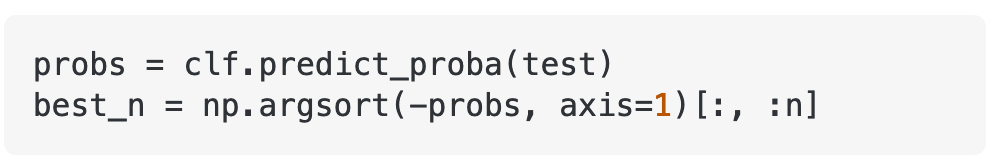
**The Classification:**

* Inherently multiclass
  + K-nearest neighbors
    - Relatively simple model
    - Finds the k nearest neighbors and guesses that it must also be what the majority of its neighbors are
  + Decision trees
    - Basically a tree of yes/no decisions
    - Prone to overfitting
  + Random forests
    - A multitude of decision trees to combat overfitting
    - Might be the way to go - works well with categorical data
  + Support vector classifier (SVC)
    - Effective in high dimensional spaces
  + Neural network
    - Try playing around with different numbers of layers and nodes within layers
      * Example on [Towards Data Science Here](https://towardsdatascience.com/machine-learning-multiclass-classification-with-imbalanced-data-set-29f6a177c1a)
  + Ensemble Methods
* Inherently binary but applied as multiclass
  + Gradient boosting
    - While not inherently multiclass, can train as one-vs-rest
  + Logistic regression
    - While not inherently multiclass, can train as one-vs-rest
    - Has capability for multinomial classification
      * See example for multinomial classifier [in github here](https://github.com/USDepartmentofLabor/Binary-Context-Transformer/blob/master/Rare%20Occupation%20Classification.ipynb)  
        
    - Linear regression is not viable since we are working with categorical datasets
  + One vs Rest -- one class is positive, one class is negative. There are n-classifiers being trained for n-classes
  + One vs One - one class is positive, one class is negative for each parking of classes. There are n\*(n-1)/2 classifiers being trained for n classes. Probability is not predicted and the given class is one with most classifications

**The Evaluation:**

* Confusion-matrices
  + Verify that accuracy is uniformly distributed
* Precision-recall curves
  + Test models
  + Precision: percentage of results which are relevant
  + Recall: percentage of total relevant results correctly classified by algorithm
* Receiver operating characteristic (ROC) curves
  + Shows two parameters: true positive rate and false positive rate
  + Works for the TP FP rates of one class, would need to make 4 for each class.
* Looking at weights of models
* F-Score
* K-Folds Cross-Validation
  + May not want to use this since our dataset is small
* Squared-error scoring
  + Standard stuff ;)

**Getting Predictions:**

* Getting top n predictions  
  
* More options on [stackoverflow](https://stackoverflow.com/questions/32461246/how-to-get-top-3-or-top-n-predictions-using-sklearns-sgdclassifier)

**Links:**

<https://scikit-learn.org/stable/modules/multiclass.html>

<https://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html>

<https://scikit-learn.org/stable/tutorial/statistical_inference/putting_together.html>

<https://medium.com/@b.terryjack/tips-and-tricks-for-multi-class-classification-c184ae1c8ffc>