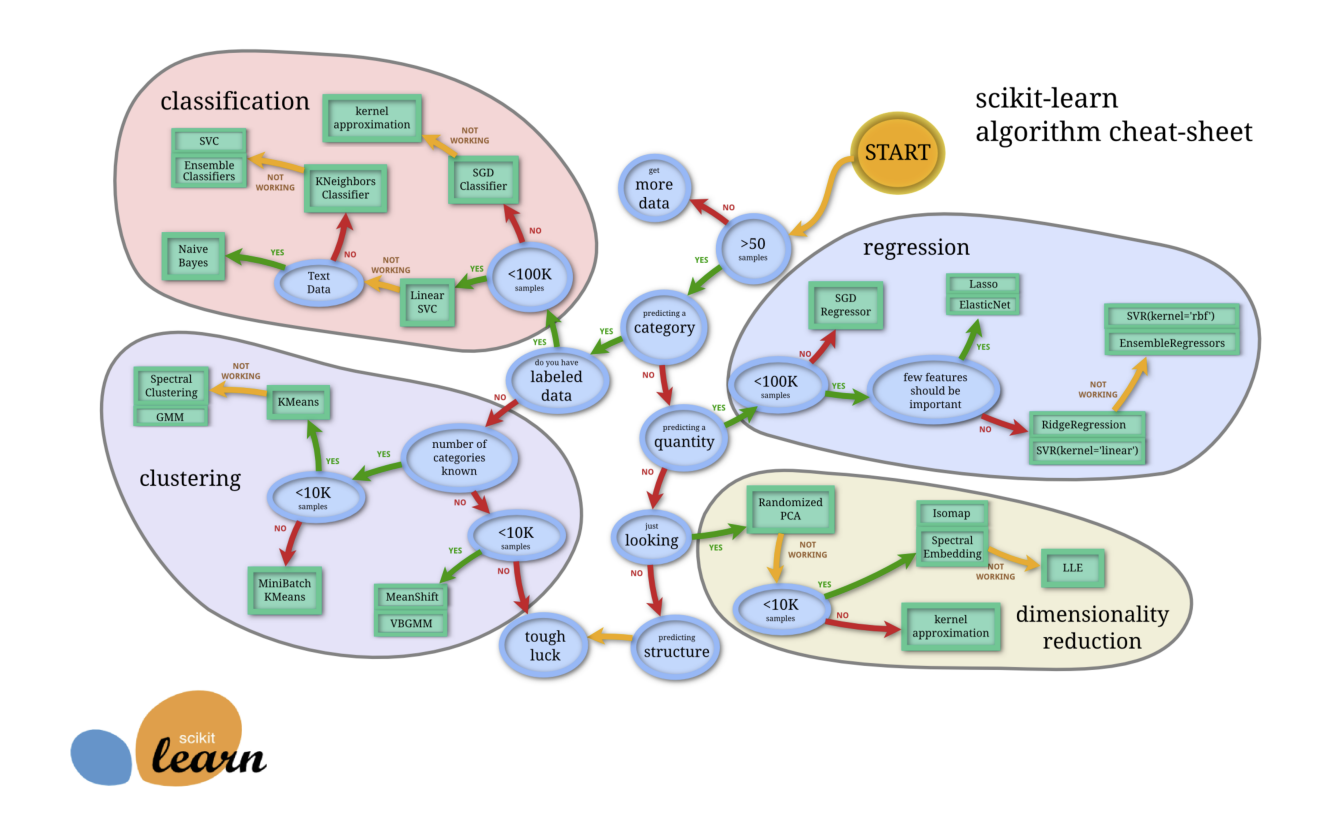
**Possible Regressors**



SGD (Stochastic Gradient Descent)

* Scikit-learn
  + Supports different loss functions (ordinary least squares, Huber loss for robust regression, linear support vector regression)
  + Well-suited for regression problems with a large number of training samples
  + Penalty determines the regularization

<https://scikit-learn.org/stable/modules/sgd.html#regression>

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDRegressor.html#sklearn.linear_model.SGDRegressor>

Lasso (Least Absolute Shrinkage and Selection Operator)

* Scikit-learn
  + Linear model with added regularization term
  + We can modify the penalty parameter alpha

<https://scikit-learn.org/stable/modules/linear_model.html#lasso>

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html#sklearn.linear_model.Lasso>

Ridge Regression

* Scikit-learn
  + Impose penalty on size of coefficients, minimize penalized residual sum of squares
  + Complexity parameter alpha controls the shrinkage
  + Larger alpha, greater shrinkage, coefficients become more robust to collinearity

<https://scikit-learn.org/stable/modules/linear_model.html#ridge-regression>

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html#sklearn.linear_model.Ridge>

SVR (Support Vector Regression)

* Scikit-learn
  + Model depends only on a subset of the training data
  + Cost function ignores samples who prediction is close to their target
  + Different implementations
    - SVR (better for smaller data sets)
    - NuSVR (use nu parameter to control number of support vectors)
    - LinearSVR (linear kernel, faster, more flexible choice of penalties and loss functions, scales better)

<https://scikit-learn.org/stable/modules/svm.html#regression>

<https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html#sklearn.svm.SVR>

<https://scikit-learn.org/stable/modules/generated/sklearn.svm.NuSVR.html#sklearn.svm.NuSVR>

<https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVR.html#sklearn.svm.LinearSVR>

Decision Trees (Regression Trees)

* Scikit-learn
  + Appears to use constant piecewise approximations
  + If max depth is set too high, decision trees learn too fine details of the training data (i.e. they start learning from the noise and overfit)

<https://scikit-learn.org/stable/modules/tree.html#tree-regression>

<https://scikit-learn.org/stable/auto_examples/tree/plot_tree_regression.html#sphx-glr-auto-examples-tree-plot-tree-regression-py> (Example)

Random Forest Regressor

* Scikit-learn
  + Fit trees to subsamples of the data
  + Use averaging to improve to improve predictive accuracy and control over-fitting

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html>

<https://scikit-learn.org/stable/modules/ensemble.html#forest>

Bayesian Ridge Regression

* Scikit-learn
  + Adapts to the data at hand
  + Can include regularization parameters in the estimation procedure
  + Inference of model can be time consuming
  + Similar to regular ridge except parameters are estimated jointly during the fit of the model
  + Using gamma prior distributions

<https://scikit-learn.org/stable/modules/linear_model.html#bayesian-ridge-regression>

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.BayesianRidge.html#sklearn.linear_model.BayesianRidge>

K-Nearest Neighbors

* Scikit-learn
  + Predicts based on interpolation among nearest neighbors in training set
  + Similar to the classification method
  + Number of neighbors to be included can be specified

<https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html>