An Introduction to Predictive Modeling in R

Ryan Benz • OC RUG • February 23, 2017

Build Something Useful!

- Predictive modeling: the process of combining data and algorithms in order to build *useful* models.

- In contrast with explicitly programming rules, predictive modeling algorithms attempt to *learn patterns from the data itself.*

- Predictive modeling has deep mathematical foundations, but in the end, it's extremely practical

Predictive Modeling is Everywhere

- Is this email message spam?
- Will this person default on their loan?
- Which other products might this person also buy?
- Is that a cat?
- Which group of people should I target for my ad campaign
- Is this person sick or health?

Lots of Contexts, Lots of Terms

- People have been predictively modeling for a long time, and in lots of different fields

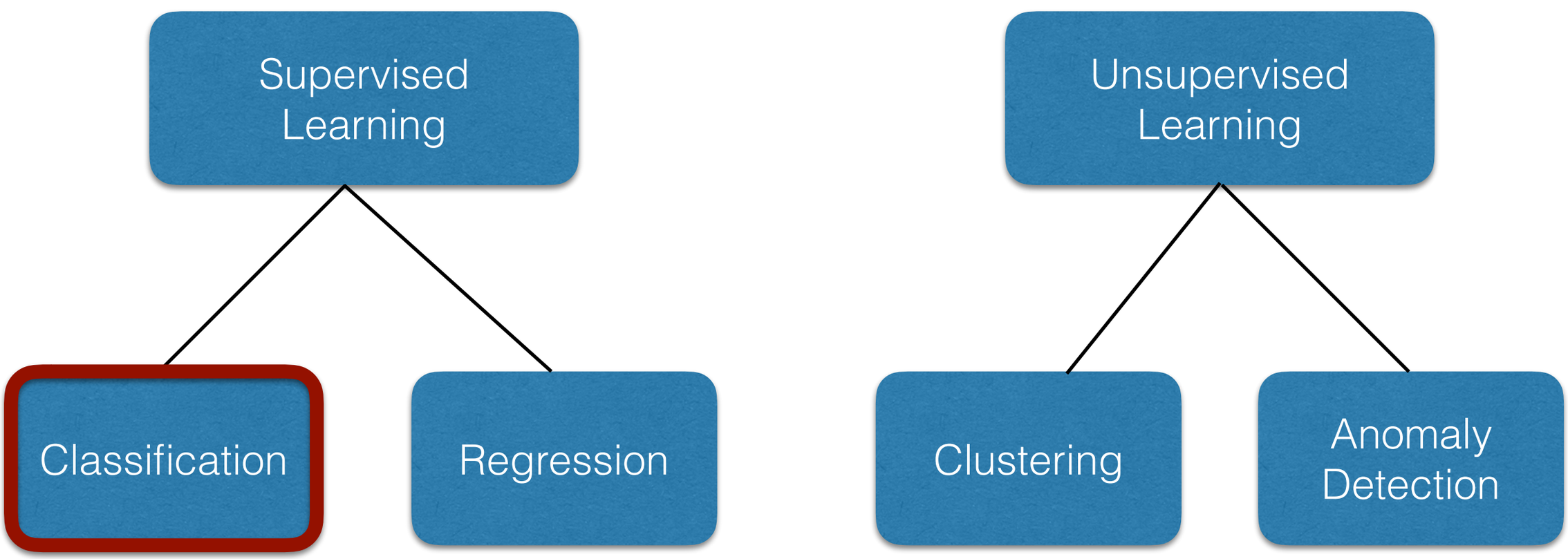
- Therefore, lots of different terms used for similar things

The Subject	**The Data**	**The Outcomes**
Predictive modeling	Features	Classes
Predictive analytics	Predictors	Labels
Machine learning	(Independent) Variables	Dependent Variables
Data mining	Measures	Responses
Statistics	Attributes	Targets

Two Branches of Machine Learning

If you have the answer for your training data

If you don't



The ^highly simplified Model Building Process

Start with a question

Which stocks should I buy?
(How can I determine good & bad stocks?)

Collect relevant data

Amass historical stock/company data
WITH good/bad calls

Train a model

Model

computery stuff

Trained Model

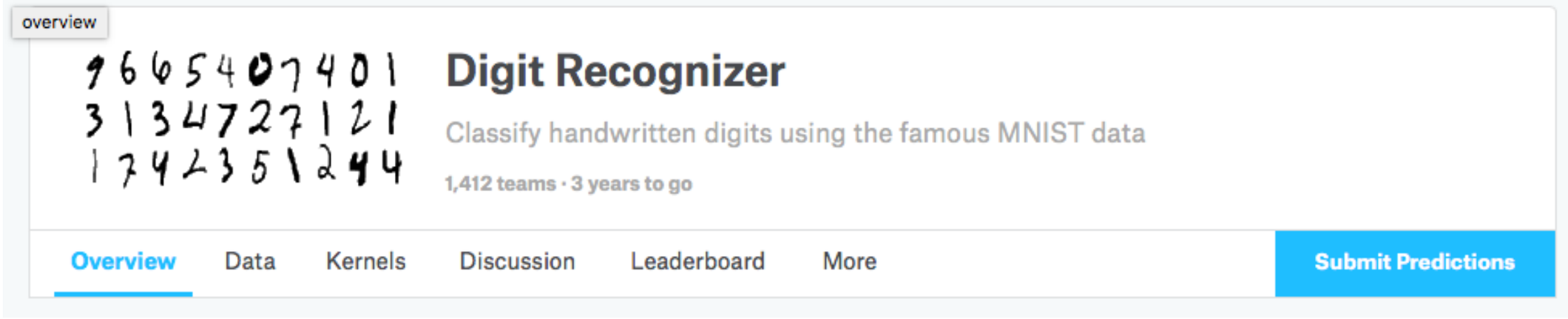
Input new data

Invest ⟶ Get RICH!!!

Good Stocks

Bad Stocks

A Real Example:
Kaggle Digit Classification Competition



Task

Given an image of a handwritten digit, determine which one it is

Training Data

A vector of length 785 for each example (digit)

- first entry is the label (a digit 0 - 9)
- the remaining 784 entries are each numbers 0 - 255 representing a 28 x 28 gray-scale image of the digit

e.g.: 3,0,0,0,27,59,82,171,201,163,74,30,0,0…0,0,0

Testing Data

A vector of length 784 for each *new* example; NO LABELS

Submission

```
ImageId,Label
1,3
2,7
3,8
(27997 more lines)
```

https://www.kaggle.com/c/digit-recognizer

A Real Example:
Kaggle Digit Classification Competition

Code

This script has been released under the Apache 2.0 open source license. **Download Code**

```r
# Creates a simple random forest benchmark

library(randomForest)
library(readr)

set.seed(0)

numTrain <- 10000
numTrees <- 25

train <- read_csv("../input/train.csv")
test <- read_csv("../input/test.csv")

rows <- sample(1:nrow(train), numTrain)
labels <- as.factor(train[rows,1])
train <- train[rows,-1]

rf <- randomForest(train, labels, xtest=test, ntree=numTrees)
predictions <- data.frame(ImageId=1:nrow(test), Label=levels(labels)[rf$test$predicted])
head(predictions)

write_csv(predictions, "rf_benchmark.csv")
```

Most of the code is about data prep!

One line to build the model, one line to make the predictions

show less

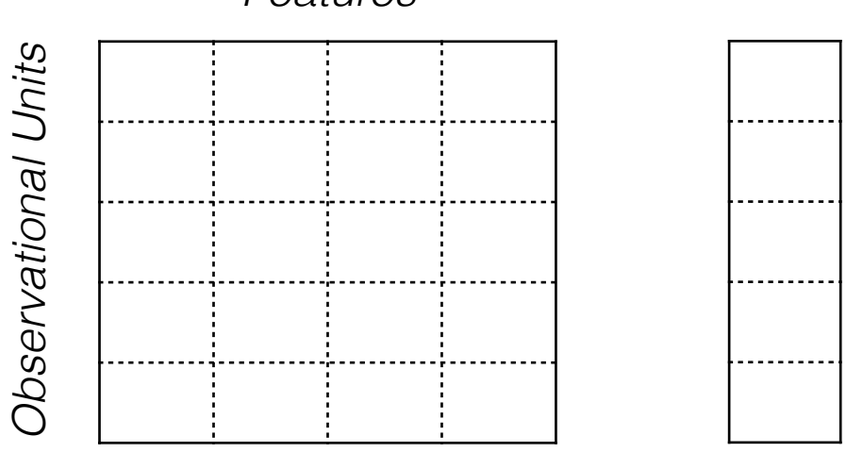
This model is 93.5% accurate

https://www.kaggle.com/c/digit-recognizer

Building Your Own Models Can Be Easy

Training Data

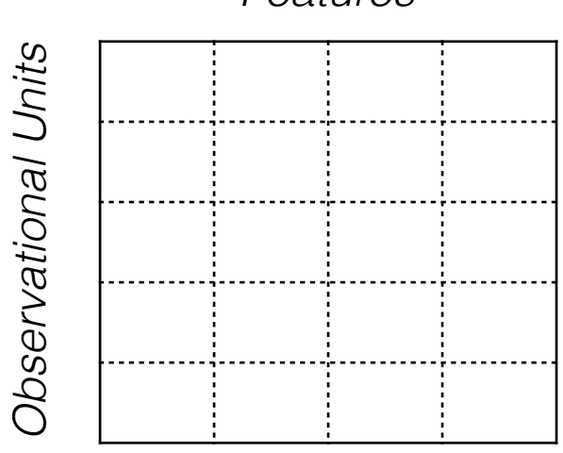
Data Matrix



Labels

Testing Data

Data Matrix



Model Training `model_func(training_matrix, training_labels, …)`

Model Predictions `predict(model_obj, testing_matrix)`

There Are Lots of Ways to Build Un-Useful Models

- Your data isn't useful for the problem you want to solve
- You model is too simple, don't have enough data (under-fitting)
- Your model is too complex, doesn't generalize (over-fitting)
- Your training data wasn't representative of the testing data
- You made a mistake somewhere

Some Thoughts About Building Predictive Models

- Ensuring your model is going to work on new, unseen data is really important
  - Is your training data representative of the new data?
  - Use resampling methods (e.g. 10-fold cross validation) to estimate the model performance

- Information “leakage” can ruin your model, is often subtle and not immediately evident; be careful

- Learning the mathematical/statistical details of various modeling algorithms and methods can be useful, but…

- Your time is often best spent understanding the problem domain, finding relevant data (once you’ve mastered the fundamentals)

- Predictive modeling is very practical, and you get good at it through lots of practice

Resources

- THE book
  Applied Predictive Modeling (Kuhn, Johnson)
  http://appliedpredictivemodeling.com

- Other books (both available online for free)
  - Elements of Statistical Learning (Hastie, et.al.)
  - Pattern Recognition and Machine Learning (Bishop)
  - Data Mining with R: Learning with Case Studies (Torgo)

- R Packages
  - 100's of modeling packages are available (e.g. `e1071`, `randomForest`, `glmnet`)
  - `caret`: addresses the entire modeling workflow

- Where to Practice
  - Kaggle (www.kaggle.com)
  - Flowing Data (https://flowingdata.com/category/statistics/data-sources/)