

# **Social Data Science**

## **Supervised Machine Learning**

Dr. Thomas Davidson

Rutgers University

November 1, 2021

# Plan

1. Course updates
2. Introduction to machine learning
3. Model evaluation
4. Classification algorithms

# Course updates

## Homework 3

- ▶ Homework 2 and Project Proposal grades and feedback released
  - ▶ Please review all comments
- ▶ Homework 3 was released on Friday and is due next Monday at 8pm
- ▶ Project Prototypes due next Friday at 5pm

# Roadmap

## A month of machine learning

1. Introduction to supervised machine learning
2. Supervised text classification
3. Challenges: Unpredictability and bias
4. Supervised image classification / computer vision

# Introduction to machine learning

## What is machine learning?

- ▶ Machine learning is a method to “automate discovery from data” (Molina and Garip 2019)
- ▶ An approach that draws upon statistical methodology and computer science
- ▶ Often referred to as “artificial intelligence”

# Introduction to machine learning

## Supervised and unsupervised learning

- ▶ *Supervised machine learning*
  - ▶ We observe an output  $Y$  for each input  $X$ .
  - ▶ The goal is to learn a function to predict  $Y$  given  $X$ ,  $Y = f(X)$
  - ▶ Often SML is used for *classification* problems, where the objective is to classify the observed data into discrete classes
    - ▶ The learning is “supervised” because we have this information
- ▶ *Unsupervised machine learning*
  - ▶ We only observe  $X$ , but there no “correct” answer  $Y$
  - ▶ The goal is typically to partition  $X$  into a set of classes, but the classes are not known in advance
    - ▶ Topic modeling is an example of an unsupervised learning algorithm

# Introduction to machine learning

## Two cultures of statistical modeling

- ▶ Consider the following linear model:

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 x + \epsilon$$

- ▶ Social scientists are typically interested in the  $\hat{\beta}$  given  $Y$ , i.e. the estimated effect of the variable  $x$
- ▶ Computer scientists are more interested in how well the model predicts  $\hat{Y}$ , paying less attention to the estimated coefficients.
- ▶ What can we learn by constructing models optimized to predict  $Y$ ?

# Introduction to machine learning

## Prediction vs. explanation

- ▶ Predictive models are specified differently to explanatory ones
  - ▶ In a regression context, we might use theory to guide the selection of a handful of *variables* to appropriately estimate  $\hat{\beta}$ .
  - ▶ In a predictive context, we want to find the function  $f(X)$ , such that  $Y = f(X)$ . This often involves many more variables and a more complex functional form.
    - ▶ Variables in the ML context are referred to as *features*.



# Introduction to machine learning

## Why predict?

- ▶ Policy interventions
  - ▶ Predictive models for targeting policies
- ▶ Document classification
  - ▶ Identifying relevant documents and classifying the content
- ▶ Record linkage and data imputation
  - ▶ Prediction can be used to merge different records and impute missing values
- ▶ Causal inference
  - ▶ New ways to estimate causal effects and uncover heterogeneity

# Introduction to machine learning

## Predictive validity

- ▶ The goal of most social scientific analyses is *explanation*
  - ▶ We develop and test theories and use data to assess how well our theoretical explanations map onto the empirical observations
  - ▶ Often little attention is paid to *prediction* and many models have very low predictive accuracy.
- ▶ Duncan Watts (2014) argues that we must pay more attention to *prediction* and consider the *predictive validity* of sociological theories.
  - ▶ This may allow us to better understand the scope and robustness of many sociological findings.
  - ▶ What good is an explanation if it isn't predictive?

See Watts, Duncan J. 2014. "Common Sense and Sociological Explanations." *American Journal of Sociology* 120 (2): 313–51. <https://doi.org/10.1086/678271> and the debate with Turco and Zuckerman.

# Introduction to machine learning

## Data splitting and model training

- ▶ In supervised machine learning, we generally split our data into two groups, *training* and *testing*
- ▶ The *training* data is used to train a model or to estimate  $Y = f(X)$ .
  - ▶ The model uses data matrix  $X$  to predict a *known*  $Y$ .
- ▶ The *testing* data is used to choose and evaluate a model. This is also referred to as *held-out* or *out-of-sample* data.
  - ▶ We take our trained model and predict  $\hat{Y}$  for the test examples.
  - ▶ We then compare  $\hat{Y}$  to  $Y$  to assess predictive accuracy.

# Introduction to machine learning

## Vignette

- ▶ Let's say we want to predict a college-attendance given information about their early childhood.
- ▶ Someone working in the explanatory framework would be to construct some regression model to predict

$$Y_{college} = \hat{\beta}_0 + \beta_{1:K} X_{1:K} + \epsilon$$

using  $K$  predictor variable, where each variable is carefully selected based on theoretical considerations

- ▶ Assuming the model is appropriately specified, we would then analyze the statistical significance of the coefficients to test our theories.
  - ▶ e.g “Mother's level of education is a positive predictor of college attendance ( $p < 0.05$ ).”

# Introduction to machine learning

## Vignette

- ▶ Now consider a predictive version of this model. Here the goal is to develop the best possible prediction of  $Y_{college}$ .
- ▶ We estimate a model using our training data,

$$Y_{college_{train}} = \hat{\beta}_0 + \beta_{1:M} X_{1:M} + \epsilon$$

- ▶ Unlike the previous model, we will include a large set of  $M$  predictor variables, where  $M \gg K$
- ▶ We then use this model to predict  $\hat{Y}_{college_{test}}$  and compare the predictions to the known values,  $Y_{college_{test}}$ .
- ▶ Finally, we make a statement about the accuracy of our model.
  - ▶ e.g. “The model predicted out-of-sample college attendance with 85% accuracy.”

# Introduction to machine learning

## Explanatory models $\neq$ predictive models

- ▶ Economists Mullainathan and Spiess (2017) evaluate the relationship between predictive and explanatory models. In an ideal world, we might want to have a model optimized for predicting  $\hat{Y}$  hats and Beta hat's.
- ▶ Explanatory models often have low-predictive power. But can predictive models be produce useful explanations?

See Mullainathan, Sendhil, and Jann Spiess. 2017. "Machine Learning: An Applied Econometric Approach." *Journal of Economic Perspectives* 31 (2): 87–106.

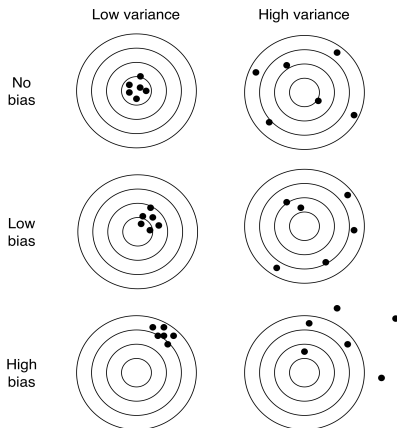
# Introduction to machine learning

## Explanatory models $\neq$ predictive models

- ▶ “The very appeal of these algorithms is that they can fit many different functions. But this creates an Achilles’ heel: more functions mean a greater chance that two functions with very different coefficients can produce similar prediction quality” (Mullainathan and Spiess 2017: 97–98).
- ▶ In short, there might be many different subsets of a dataset that produce equally good predictions. This makes it hard to develop a coherent explanation based on a predictive model.

# Model evaluation

## Bias-variance trade-offs



Salganik 2017. See [this website](#) for a visualization of the bias-variance trade-off.



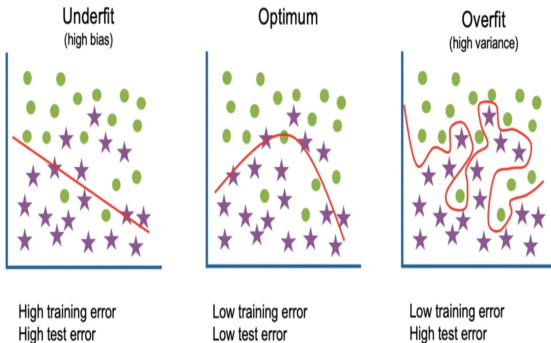
# Model evaluation

## Underfitting and overfitting

- ▶ *Underfitting* occurs when a model poorly fits the data.
  - ▶ e.g. A linear model may not capture non-linear relationships between variables.
- ▶ *Overfitting* occurs when a model fits random noise in the training data.
  - ▶ If a model has overfit then it does not generalize well to unseen data.
  - ▶ This tends to be a more serious problem in machine-learning than underfitting since we often have richly parameterized models

# Model evaluation

## Underfitting and overfitting



# Model evaluation

## Out-of-sample validation

- ▶ *Out-of-sample validation* is used to directly measure over/underfitting, something generally ignored in explanatory approaches to statistics (Watts 2014).
  - ▶ An underfit model will perform poorly both in and out-of-sample
  - ▶ An overfit model will perform well in-sample but poorly on unseen data
- ▶ The main challenge is to estimate a model that will generalize well to unseen data without learning too much about idiosyncratic variance in the training data

# Model evaluation

## Cross-validation

- ▶ Train-test splits reduce the amount of data available to us, increasing risk of underfitting and potentially making results sensitive to the particular split.
- ▶ *Cross-validation* is an approach to split the data into small test-train subsets.
- ▶ A popular approach is *k-fold* cross-validation where we split the data into  $k$  subsets.
  - ▶ We successively train a model on each  $k - 1$  folds and test it on the  $k^{th}$  fold.
  - ▶ The results are then averaged across all  $k$  folds.

# Model evaluation

## Cross-validation



Source: Wikipedia.

# Model evaluation

## Cross-validation

- ▶ The extreme is called *leave-one-out* (LOO) cross-validation.
  - ▶ Given  $N$  observations, we train  $N$  models, each time using  $N - 1$  data points.
  - ▶ This is rarely used on large datasets because it is very computationally expensive, although variations are common in Bayesian statistics.

'This technique is similar to a technique used in conventional statistics known as *jackknife* resampling.

# Model evaluation

## Regularization

- ▶ Machine learning techniques are often prone to overfitting because the models are complex with many parameters
- ▶ *Regularization* is another approach we can use to prevent overfitting.
  - ▶ We constrain the parameter space by removing some complexity to try to prevent the model fitting “noise” in the data
- ▶ Most machine learning algorithms have regularization procedures

# Model evaluation

## Regularization

- ▶ The *least absolute shrinkage and selection operator* (LASSO) imposes a penalty on regression coefficients
  - ▶ A regression model finds the least squares estimate, minimizing the squared error of the predictions  $\sum_{i=1}^n (y_i - \bar{y})^2$
  - ▶ LASSO includes an additional “penalty” term, minimizing  $\lambda \sum_{j=1}^k |\beta_j|$ , the sum of the absolute values of the regression coefficients
    - ▶ Here  $\lambda$  is a coefficient controlling the strength of the regularization
  - ▶ In practice, this forces many coefficients to equal zero,  $\beta_j = 0$ , reducing the complexity of the parameter space by ignoring many variables
- ▶ Many *neural network* models often do something similar, randomly setting a subset of parameters to zero each iteration (“drop out”)



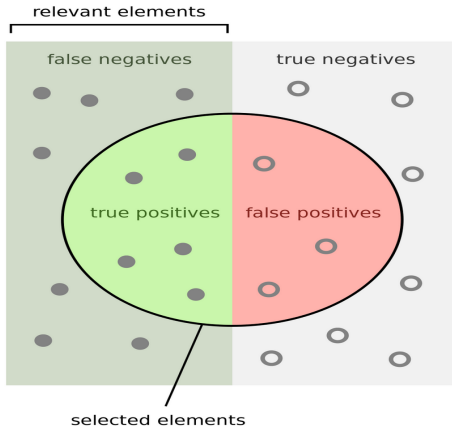
# Model evaluation

## Metrics: Binary classification

- ▶ The following metrics apply to binary classification problems, although many can be generalized to multi-class or continuous outcomes.
- ▶ A binary classifier learns a function  $f(X)$  to predict  $Y$ , where  $Y = 1$  or  $Y = 0$ .
  - ▶ Many algorithms return a predicted probability  $P(Y|X)$ , but some only return a discrete value (1 or 0).

# Model evaluation

Metrics: TP, FP, TN, FN




Source: Wikipedia.

# Model evaluation

## Metrics: Precision (TP/TP+FP)

How many selected items are relevant?


$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$


Source: Wikipedia.

# Model evaluation

Metrics: Recall (TP/TP+FN)

How many relevant items are selected?

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$


Source: Wikipedia.

# Model evaluation

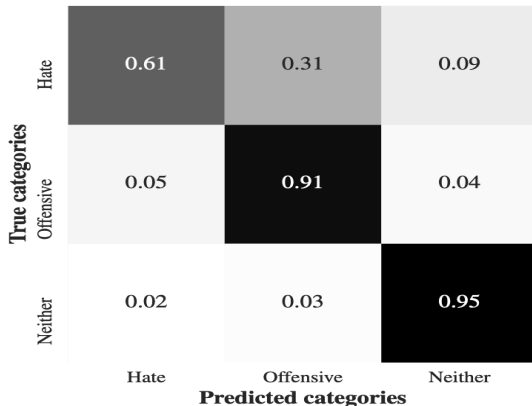
## Metrics: F1

The  $F_1$  score is the *harmonic mean* of precision and recall and is often used as an overall description of predictive performance for a classifier.

$$F_1 = 2 \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

# Model evaluation

## Metrics: The confusion matrix



Davidson, Thomas, Dana Warmley, Michael Macy, and Ingmar Weber. 2017. "Automated Hate Speech Detection and the Problem of Offensive Language." In Proceedings of the 11th International Conference on Web and Social Media (ICWSM), 512–15.

# Model evaluation

## Metrics: Receiver Operating Characteristic (ROC) curve

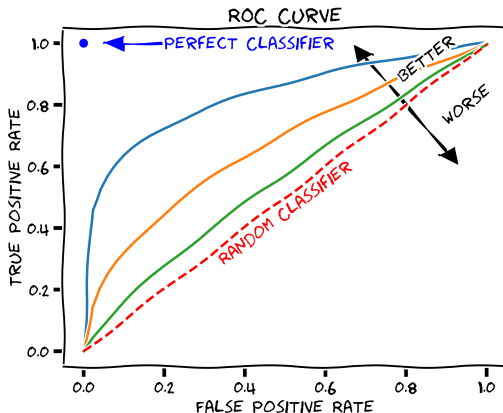
- ▶ If an algorithm returns a predicted probability then we must identify a *threshold* for class assignment
- ▶ In the binary case,

$$\text{Class}(Y) = \begin{cases} 1, & P(\hat{Y}|X) \geq \text{threshold} \\ 0, & P(\hat{Y}|X) < \text{threshold} \end{cases}$$

- ▶ Plot the true positive rate ( $TPR = TP / TP + FN$ ) against false positive rate ( $FPR = FP / FP + TN$ ) for different predicted probability thresholds to identify the optimal value. This is known as the *ROC* curve.
- ▶ The area under the ROC curve (*AUC*) provides an overall measure of classifier performance.

# Model evaluation

## Metrics: Receiver Operating Characteristic (ROC) curve



Source: Wikipedia



# Model evaluation

## Metrics

- ▶ The choice of metric depends on the outcome of interest and what you want to optimize for. Often we might want to use a metric like ROC or F1 to find a compromise.
- ▶ Consider a carbon monoxide alarm:

	Alarm	No alarm
CO present	TP	FN
CO absent	FP	TN
- ▶ False negatives are really bad and should be avoided at all costs.
- ▶ Too many false positives will also be bad, as it may lead people to remove the batteries from the alarm, but a low level will be tolerated.

# Classification algorithms\*

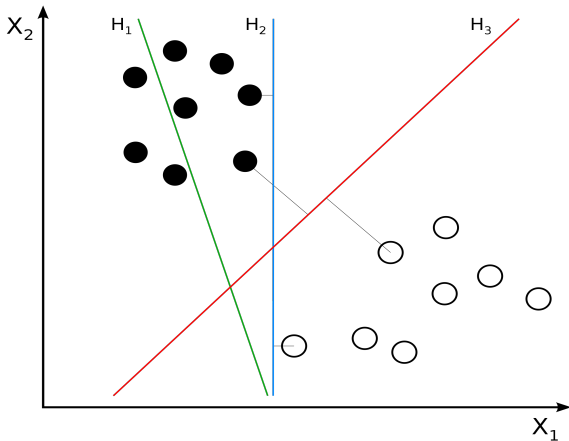
## Logistic regression

- ▶ Logistic regression is a regression model for binary outcomes (although there is some debate about when we should estimate a standard linear probability model using OLS).
- ▶ Uses a logit function to estimate the log-odds of an event ( $Y = 1$ ) given predictors  $X$ .
- ▶ Multinomial logistic regression can be used if you have a multi-class outcome.
  - ▶ e.g. A model predicting level of education.

\*Many of these algorithms can also be used for regression problems where the outcome is continuous.

# Classification algorithms

## Support Vector Machines

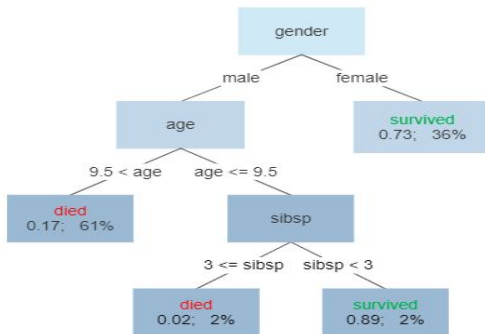


Source: Wikipedia

# Classification algorithms

## Decision Trees

### Survival of passengers on the Titanic



Source: Wikipedia. See this website for an excellent visual introduction to decision trees.

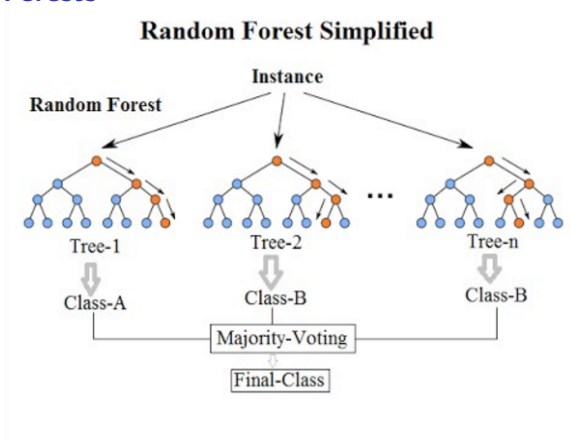
# Classification algorithms

## Random Forests

- ▶ Decision trees tend to overfit the training data
- ▶ Solution: Grow lots of trees and average over them
  - ▶ Using a procedure called *bootstrap aggregating* or *bagging* for short we can sample from our data and generate a *forest* consisting of many decision trees. This is known as an *ensemble* method because it involves more than one model.
  - ▶ The approach is effective because the algorithm randomly splits the data into leaf nodes based on different features, hence it is a *random* forest.
  - ▶ The final classification is an average across the different decision trees.

# Classification algorithms

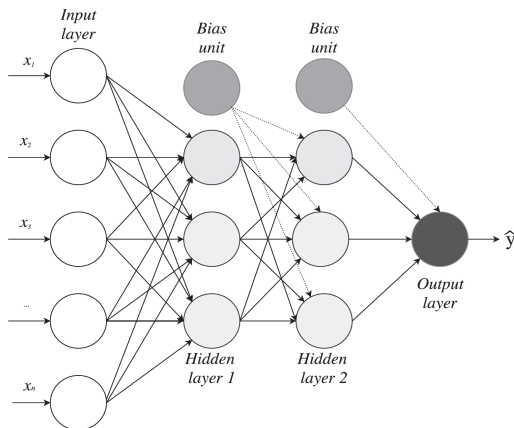
## Random Forests



Source: Wikipedia

# Classification algorithms

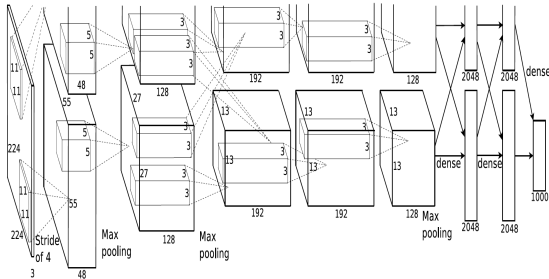
## Neural networks



Davidson 2019.

# Classification algorithms

## Neural networks



Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. 2012. "Imagenet Classification with Deep Convolutional Neural Networks." In *Advances in Neural Information Processing Systems*, 1097–1105.



# Classification algorithms

## Hyperparameters

- ▶ Each algorithm has hyperparameters that can adjust how it works.
  - ▶ e.g. Regularization type for logistic regression and SVM.
  - ▶ e.g. Number of trees, tree depth, and splitting criterion for random forest.
  - ▶ e.g. Number of layers, activation function, and optimization routine for neural networks.
- ▶ Often we want to find the algorithm that best fits the data so we conduct a search over different hyperparameters and compare many different models.
  - ▶ In many cases we also want to test the effect of different pre-processing or feature construction steps.

# Classification algorithms

## Hyperparameter search and computational complexity.

- ▶ Davidson (2019) uses neural network models to predict high school GPA.
  - ▶ Three model hyperparameters with 40 different combinations
    - ▶ Number of hidden layers (depth)
    - ▶ Number of neurons per hidden layer (breadth)
    - ▶ Activation function
  - ▶ Each model is trained using 5-fold cross-validation, resulting in 200 different model fits
- ▶ These models took over 12 hours to estimate on a high-end Macbook Pro.

Python code and output is available [here](#).

# Classification algorithms

## Black-box models and interpretability

- ▶ In contrast to standard explanatory models, which are considered to be interpretable, many of these algorithms are described as “black boxes,” meaning that we are unable to observe their workings.
- ▶ There is a trade-off between model complexity (often associated with better predictions) and human interpretability
  - ▶ Watts (2014) argues that it may be worth sacrificing some interpretability in the interest of better predictions.
- ▶ But there are lots of developments in the field of ML interpretability
  - ▶ See Christoph Molnar’s open-source book *\*Interpretable Machine Learning\** for an overview.

# Classification algorithms

## Black-box models

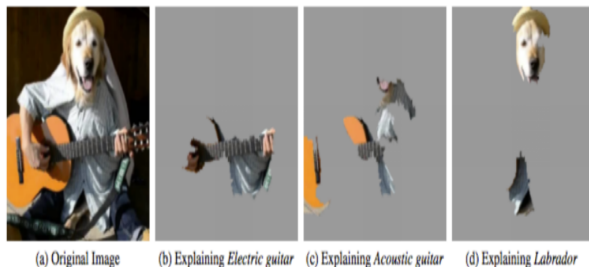
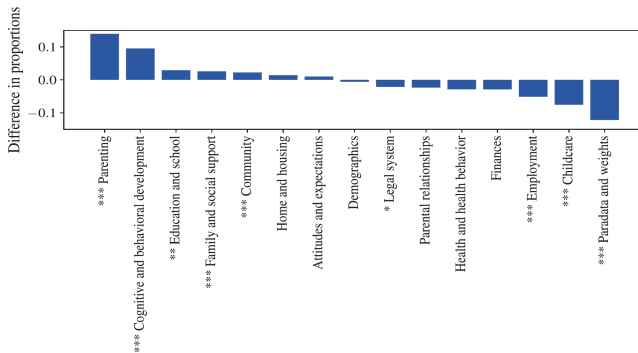


Figure 4: Explaining an image classification prediction made by Google's Inception network, highlighting positive pixels. The top 3 classes predicted are "Electric Guitar" ( $p = 0.32$ ), "Acoustic guitar" ( $p = 0.24$ ) and "Labrador" ( $p = 0.21$ )

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. " 'Why Should I Trust You?': Explaining the Predictions of Any Classifier." In Proceedings of the 22nd ACM SIGKDD, 1135–44. ACM. <https://doi.org/10.1145/2939672.2939778>.

# Classification algorithms

## Black-box models



Umbrella topic

Davidson 2019.

# Summary

- ▶ Machine learning techniques allow us to “automate discovery from data”
- ▶ Supervised and unsupervised ML techniques
- ▶ Prediction vs. explanation
- ▶ Over and under-fitting
- ▶ Regularization
- ▶ Commonly used algorithms