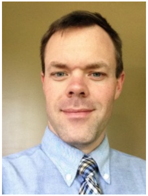
Principles of Prediction and Inference in Machine Learning
Part 1: Principles of Supervised Learning

Matthew S. Shotwell, Ph.D.

Department of Biostatistics
Vanderbilt University Medical Center
Nashville, TN, USA

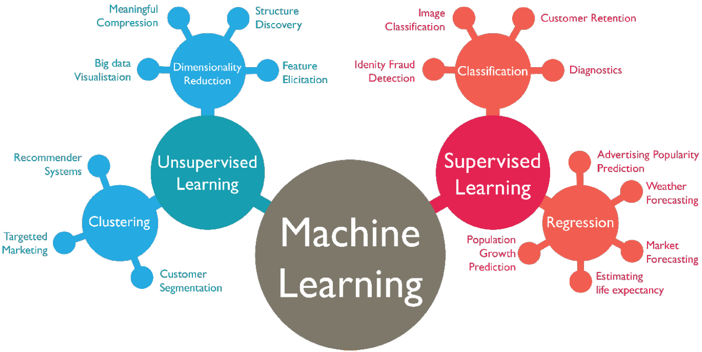
February 17, 2021

My Bio



- Matthew (Matt) S. Shotwell, Ph.D.
- Assoc. Prof. in Biostatistics
- 10 years at VU/VUMC
- R user 10+ years
- Teach "Statistical Machine Learning"; 6 years

Machine learning

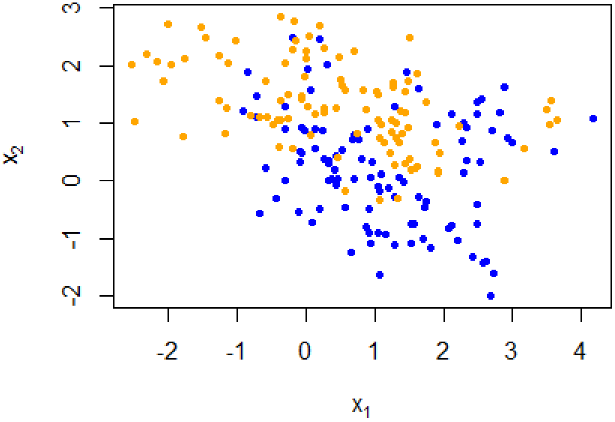


source: https://www.wordstream.com/blog/ws/2017/07/28/machine-learning-applications

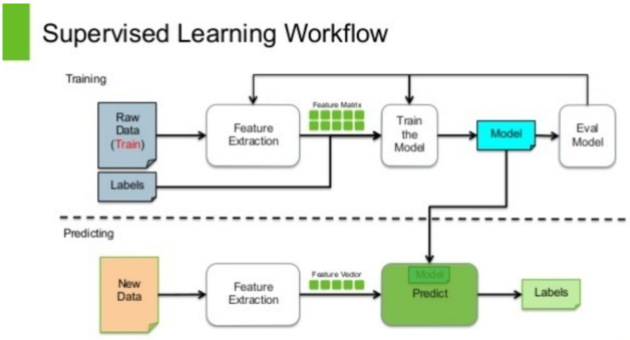
Supervised learning (Prediction)

- Have input ('features') AND output ('target')
- Create a model ('learner') using observed inputs and outputs
- Goal is to predict outputs from new inputs
- "Supervised" because we have inputs *and outputs*

Supervised learning example



Supervised learning workflow



source: https://www.quora.com/What-is-pattern-recognition

Definitions: variable types

- quantitative - e.g. blood pressure
- qualitative - e.g. gender, a.k.a. categorical, discrete, factor, numeric codes for qualitative variables called ‘targets’
- ordered - e.g. numerical pain scale (0-10)

Definitions: supervised learning tasks

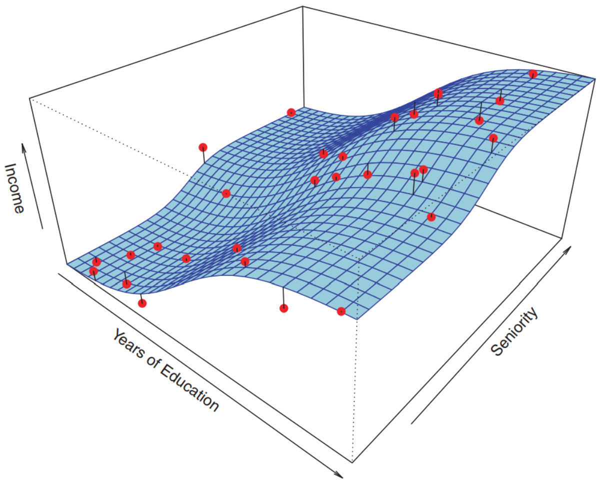
- regression - model to predict quantitative output
- classification - model to predict qualitative output
- regression or classification - ordered output

Definitions: notation

- inputs - X
- quantitative outputs - Y
- qualitative outputs - G
- transpose - X^T
- prediction - $\hat{Y}$

Regression

- Find function f to predict Y: $\hat{Y} = \hat{f}(X)$



source: ISLR Fig. 2.3

Regression

- To find f in $\hat{Y} = \hat{f}(X)$, need an **objective function**:
  - Objective function must make sense in context of the SL task
  - Least squared error: minimize $1/n \sum_{i=1}^{n} (y_i - f(x_i))^2$
  - Least absolute error: minimize $1/n \sum_{i=1}^{n} |y_i - f(x_i)|$
- and need to select a class of models $f(X)$:
  - linear models:

$$f(X) = X\beta$$

  - k-nearest neighbor:

$$f(X) = \frac{1}{k} \sum_{x_i \in N_k(X)} y_i$$

Least-squares regression (LS)

- Given input X, predict Y ($n \times 1$ matrix) as follows:

$$\hat{Y} = \hat{f}(X) = X\hat{\beta}$$

  where $\hat{\beta}$ is the value that minimizes the sum of squared error in the training data
- The R function 'lm' will estimate β in this way

k-nearest neighbor regression (kNN)

- Predict $\hat{Y}$ corresponding to X by averaging the Y values of the k nearest neighbors to X:

$$\hat{Y}(X) = \frac{1}{k} \sum_{x_i \in N_k(X)} y_i$$

- $N_k(X)$ is set of k training inputs nearest to X, as determined by a distance metric, e.g., the Euclidean distance
- k parameters is a 'smoothing parameter' or 'tuning parameter'
- 'caret::knnreg' implements kNN regression with Euclidean dist.

LS vs. NN method: flexibility and tuning

- LS is a parametric method; no tuning parameter
- NN is a semiparametric method; tuning parameter k
- LS cannot automatically model complex associations
- NN can automatically model complex associations
- LS can be made more flexible, i.e "tuned", by making the linear predictor more flexible, using 1) more predictors, 2) predictor interactions, and 3) nonlinear transformations of predictors (e.g., splines)

Bias-variance tradeoff

- more model flexibility (e.g., small k for kNN mehtod) results in less bias, more variance in $\hat{Y}$
- bias and variance are interpreted across training samples
- this is called the **"bias-variance" tradeoff**

Classification (binary case)

- Think of $\hat{Y}$ as the probability of a '1' outcome
- The predicted class $\hat{G}$ is '1' if $\hat{Y} > 0.5$ and '0' otherwise
- Can often estimate $\hat{Y}$ in a manner similar to regression (e.g., LS and NN).

Classification example

- Y- qualitative (binary) outcome (orange - 1, blue - 0)
- X_1 - quantitative predictor
- X_2 - quantitative predictor
- classification rule: $\hat{G} =$ orange if $\hat{Y} > 0.5$ else blue

Linear classification

Elements of Statistical Learning (2nd Ed.) ©Hastie, Tibshirani & Friedman 2009 Chap 2

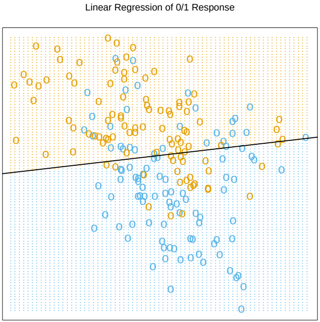


FIGURE 2.1. *A classification example in two dimensions. The classes are coded as a binary variable (*`BLUE` $= 0$, `ORANGE` $= 1$*), and then fit by linear regression. The line is the decision boundary defined by* $x^T\hat{\beta} = 0.5$. *The orange shaded region denotes that part of input space classified as* `ORANGE`*, while the blue region is classified as* `BLUE`.

k-nearest neighbor classification

Elements of Statistical Learning (2nd Ed.) ©Hastie, Tibshirani & Friedman 2009 Chap 2

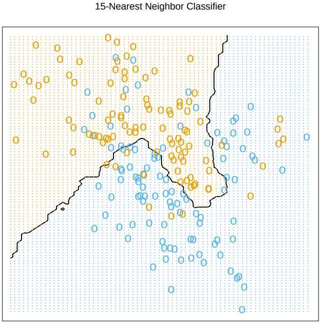


FIGURE 2.2. *The same classification example in two dimensions as in Figure 2.1. The classes are coded as a binary variable* (BLUE $= 0$, ORANGE $= 1$) *and then fit by* 15*-nearest-neighbor averaging as in (2.8). The predicted class is hence chosen by majority vote amongst the* 15*-nearest neighbors.*

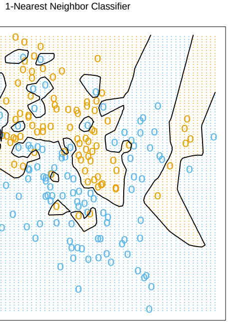


FIGURE 2.3. *The same classification example in two dimensions as in Figure 2.1. The classes are coded as a binary variable* (BLUE $= 0$, ORANGE $= 1$)*, and then predicted by* 1*-nearest-neighbor classification.*

Evaluating the model and 'optimism'

- Need a mechanism to evaluate predictive quality of model, by comparing predictions to observed targets: $\hat{Y} = \hat{f}(X)$ vs Y.
- The **prediction error** of a trained model may be evaluated using the same objective function for estimation (e.g., mean of squared errors: $1/n \sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$), but it may be quantified in some other way, e.g., AUROC for classification problems.

Evaluating the model and ‘optimism’

- The **training error** is the prediction error computed using the training data; training error is optimistic, it should not be used to select a model or tuning parameters
- The **test error** is the prediction error computed on new data (“testing data” “out of sample data”); test error is not optimistic, can be used to selet model or tune parameters
- The **optimism** is the difference in training and test error

Model tuning

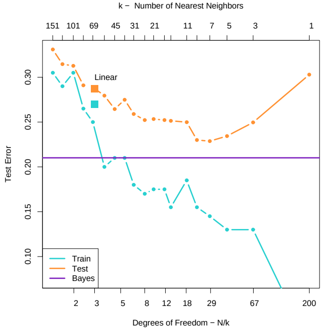


FIGURE 2.4. *Misclassification curves for the simulation example used in Figures 2.1, 2.2 and 2.3. A single training sample of size* 200 *was used, and a test sample of size* 10,000*. The orange curves are test and the blue are training error for k-nearest-neighbor classification. The results for linear regression are the bigger orange and blue squares at three degrees of freedom. The purple line is the optimal Bayes error rate.*

Model tuning

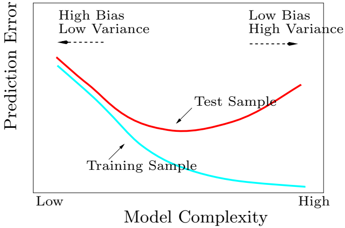


FIGURE 2.11. *Test and training error as a function of model complexity.*

Estimating test error: testing/training split

- Split data into training and testing data
- Use training data to build models
- Use testing data to evaluate models
- Simple, but not most efficient use of data

Estimating test error: k-fold cross validation

- Split data into k subsets or 'folds'
- Use k-1 subsets to build model
- Use holdout subset to evaluate model
- Repeat for all k permutations
- Results in k estimates of test error; use mean and sd
- More complicated, but also more efficient use of data

Other supervised learning methods

- the principles we discussed can be applied when using any type of supervised learning method or algorithm
  - supervised learning workflow
  - selecting a suitable objective function
  - training error, test error, optimism
  - process of tuning/selecting models by minimizing test error

Other supervised learning methods

- other supervised learning methods include
  - linear regression/classification methods
    - regularization (ridge, lasso)
    - kernel methods/local regression
    - basis function methods (e.g., splines)
  - support vector maching (classification)
  - classification and regression trees (CART)
  - bagging and boosting methods (e.g., random forest)
  - neural networks and deep learning

Prediction vs. Inference

Generally two types of tasks:

- Prediction: Is this a picture of a cat or a dog?
- Inference: Does greenhouse gas affect global average temperature?

Prediction vs. Inference

Generally two types of tasks:

- Prediction: Predicting an outcome.
- Inference: Making inferences about an unknown structure, mechanism, or relationship: e.g., effect of an exposure on an outcome.
- Need to ask: What is my primary task?
- Need to ask: Can I do both at the same time?
- Need to ask: What to consider when doing prediction/inference?

Breakout session (at 5pm)

- In my breakout session, we will work through the k-NN example in R using the orange/blue classification data. We will go through the model fitting and tuning steps.