IST 3420: Introduction to Data Science and Management

Langtao Chen, Fall 2017

8. Predictive Analytics

Learning Objectives

- Understand the concept of predictive analytics and predictive modeling process
- Understand under-fitting and over-fitting of predictive models
- Understand predictive model evaluation methods such as simple split, cross-validation, and leave-one-out
- Be able to use caret R package to facilitate predictive analytics
- Understand prediction and classification methods such as regression, k-NN, naïve Bayes, neural nets, SVM, and ensembles
- Be able to apply various predictive analytics methods to solve real problems

AGENDA

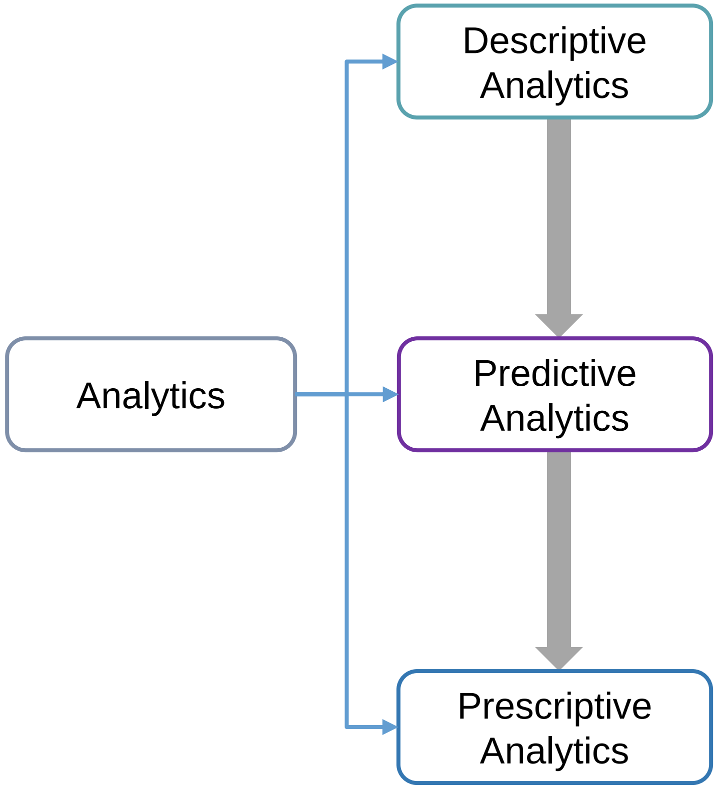
- Introduction to Predictive Analytics
- Predictive Performance Evaluation
- Using caret R Package
- Case Study: Predict Customer Churn
- Prediction and Classification Methods

The Emergence of Predictive Analytics

- Ever-increasing data available for decision making
  - Accumulated data in databases or data warehouses
  - Huge amount of data generated by sensors

- Availability of cost-efficient computation power

Recap: Overview of Analytics



What happened?

- Prepares and analyzes ***historical*** data
- Identifies patterns from samples for reporting of trends

- Visualization
- Report
- Dashboard
- …

What will happen?

- Predicts future probabilities and trends
- Finds relationships in data that may not be readily apparent with descriptive analysis

- Data mining
- Forecasting
- Statistical analysis
- …

How can we make it happen?

- Evaluates and determines new ways to operate
- Targets business objectives
- Balances all constraints

- Optimization
- Simulation
- Decision modeling
- …

The Institute for Operations Research and the Management Sciences (INFORMS) is the largest society in the world for professionals in the field of operations research (O.R.), management science, and analytics.

Source: https://www.informs.org/Community/Analytics

What is Predictive Analytics?

Definition by SAS

Predictive analytics is the use of data, statistical algorithms and machine-learning techniques to identify the likelihood of future outcomes based on historical data.

- An emphasis on prediction (rather than description, or clustering)
- Rapid analysis measured in hours or days (rather than the stereotypical months of traditional data mining)
- An emphasis on the business relevance of the resulting insights (no ivory tower analyses)
- (increasingly) An emphasis on ease of use, thus making the tools accessible to business users

http://www.sas.com/en_us/insights/analytics/predictive-analytics.html
http://www.gartner.com/it-glossary/predictive-analytics

Two Types of Predictive Analytics

- Prediction: to predict a continuous variable
  - How many items will be sold in the next month?
  - What will be the average house price in Rolla in the next year?
  - ……
- Classification: to classify units into categories
  - Which brand will be purchased?
  - Will the consumer buy the product or not?
  - Will the account holder pay off or default on the loan?
  - Is this bank transaction true or fraudulent?
  - ……

Application of Predictive Analytics: A Case in Insurer Industry

Insurance Industry Use of Predictive Analytics	
Marketing	Property-casualty insurers can use predictive analytics to analyze the purchasing patterns of insurance customers. This information can be used to increase the marketing function's *hit ratio* and *retention ratio*.
Underwriting	Insurers can use predictive analytics to filter out applicants who do not meet a pre-determined model score. This type of screening can greatly increase an insurer's efficiency by reducing the employee hours it may have spent researching and analyzing an applicant who ultimately is not a desired insured. If an applicant's model score is sufficient for consideration, then the model score can be used as a rating mechanism on which the insurer can base a variety of price/product points.
Claims	Insurers can use predictive analytics to help identify potentially fraudulent claims. It also can be used to score claims based on the likely size of the settlement, enabling an insurer to more efficiently allocate resources to higher priority claims.

Source: Charles Nyce, "Predictive Analytics White Paper"

Some Use Cases of Predictive Analytics

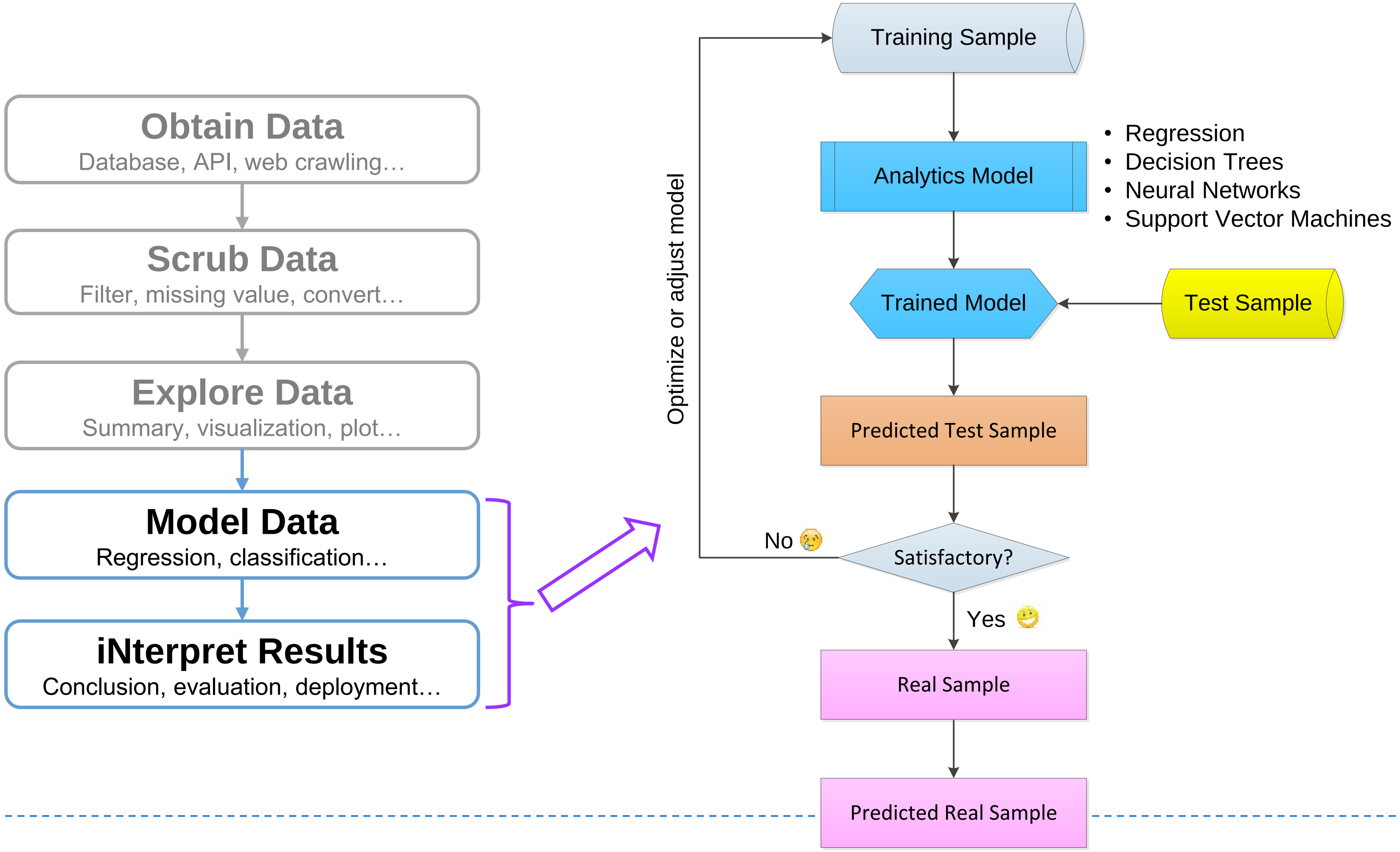
- Churn Prevention
  - Identify those customers or customer segments that are at the most risk for leaving
- Customer Segmentation
  - Identify target markets based on real data and indicators
- Product recommendation
  - Recommend books, movies, and songs to target customers
- Equipment Maintenance
  - Predict both timelines for probable maintenance events and upcoming capital expenditure requirements
- Supply Chain
  - Predict customer demand to reduce inventory and logistics cost
- Reputation Analysis
  - Predict organization’s reputation from customer feedback and posts
- …

Disadvantages of Predictive Analytics

- Requirement on the quantity and quality of data
- Inherent inaccuracy of the predictive model
- Resistance to change old operating procedures in the organization
- Investment on hardware and software of the analytics platform

Predictive Modeling Process

- Predictive modeling is a process used in predictive analytics to create a statistical model of future behavior.

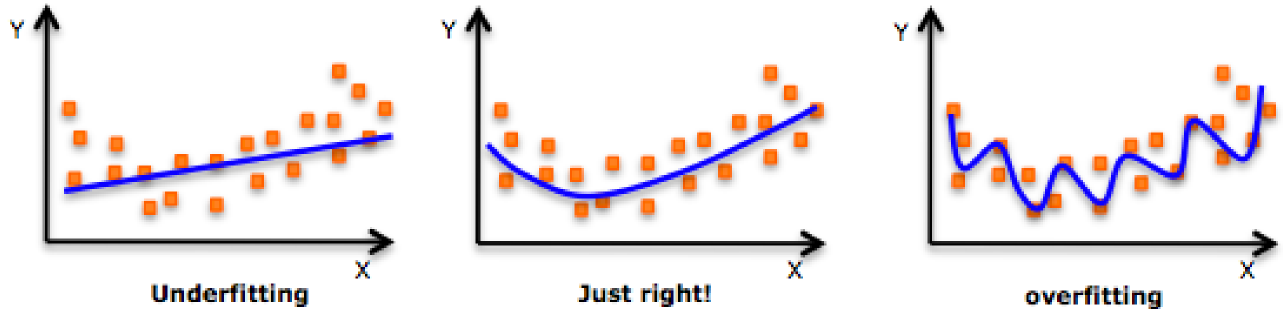


Agenda

- Introduction to Predictive Analytics
- Predictive Performance Evaluation
- Using caret R Package
- Case Study: Predict Customer Churn
- Prediction and Classification Methods

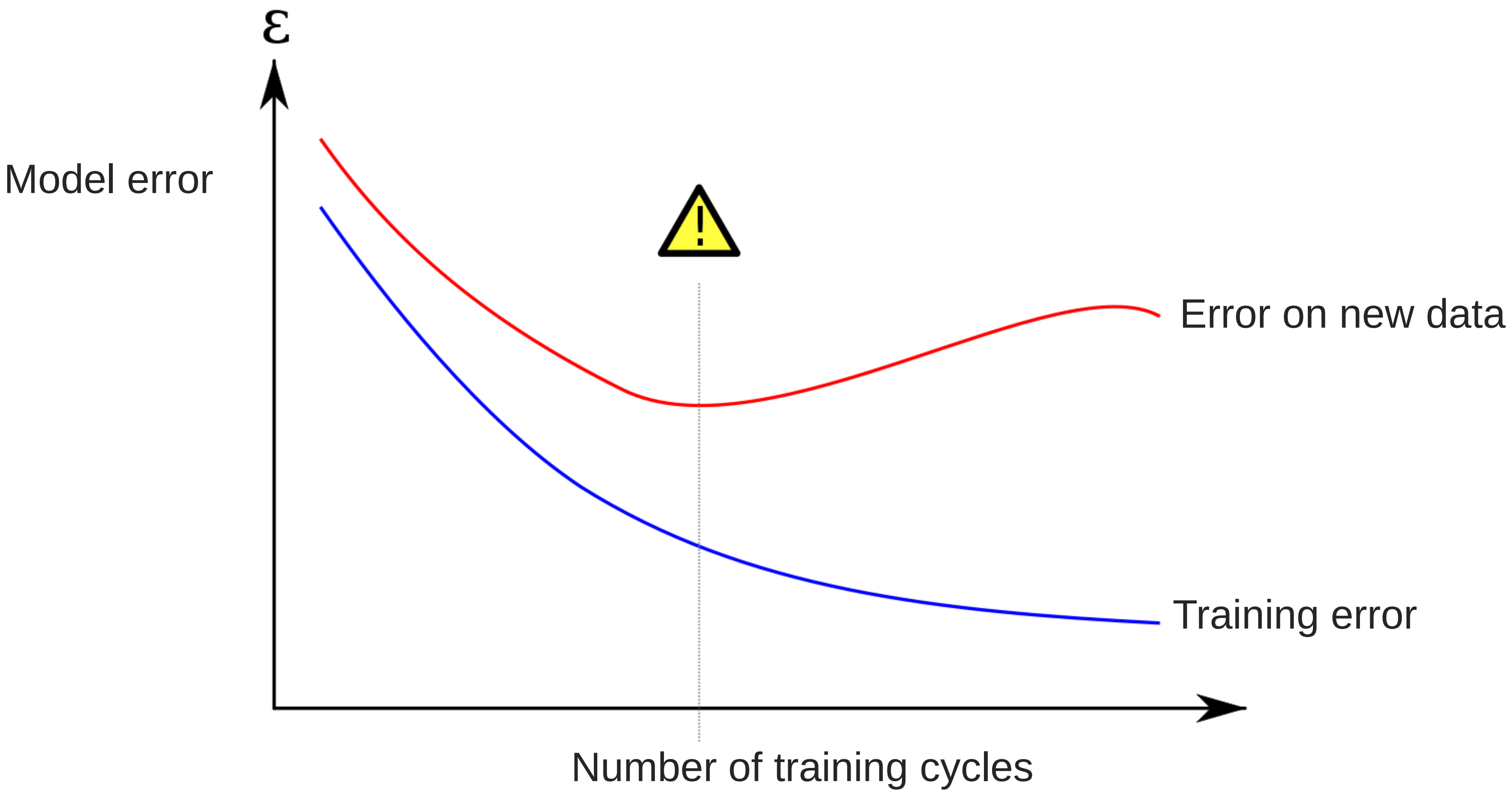
Under-fitting vs. Over-fitting

- Under-fitting
  - The model performs poorly on the training data.
  - The model is unable to capture the relationship between predictors and the response.
- Over-fitting
  - The model performs well on the training data but poorly on the test data.
  - The model is unable to generalize to unseen cases.

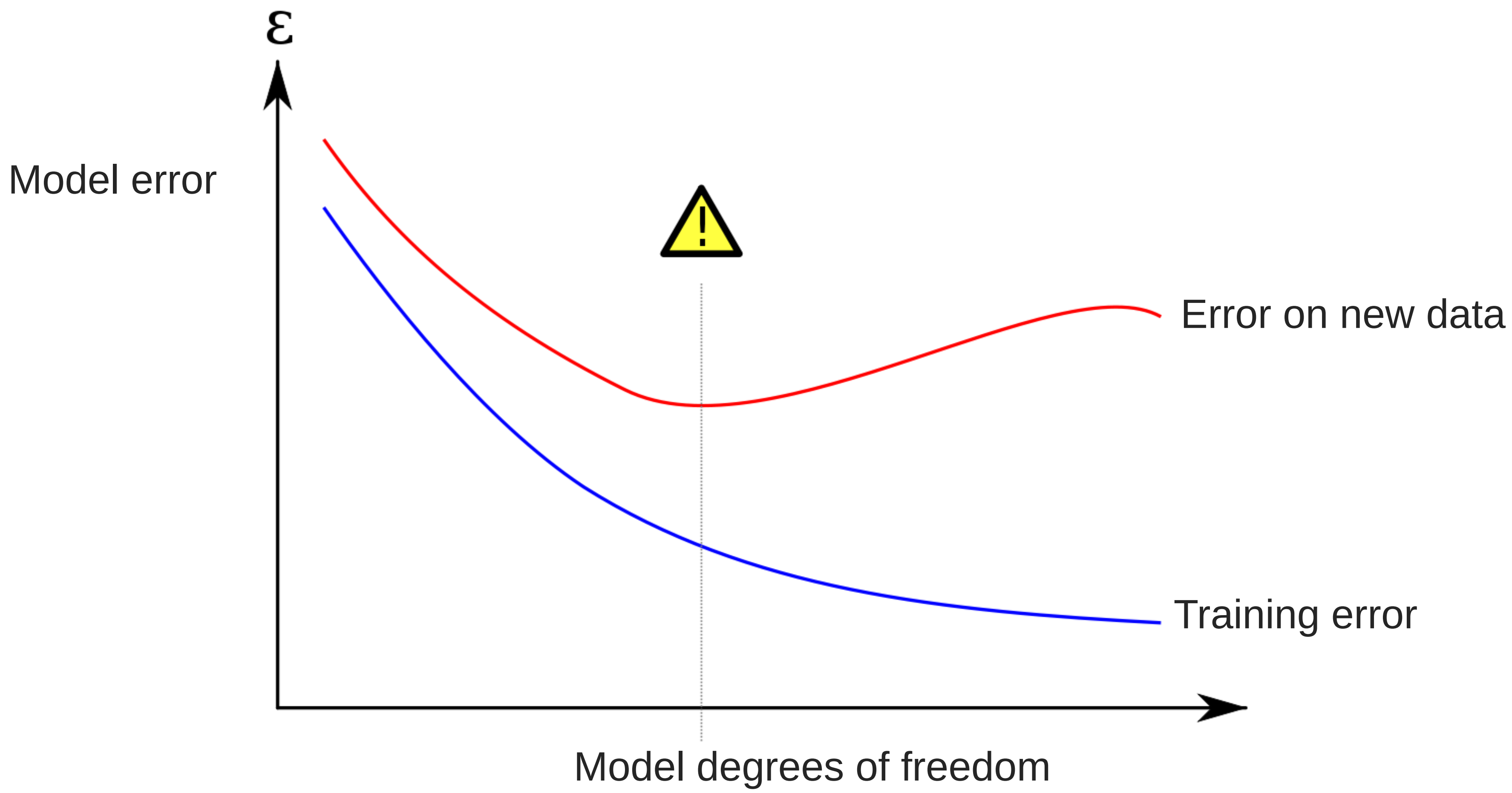


Model Over-fitting Due to Training

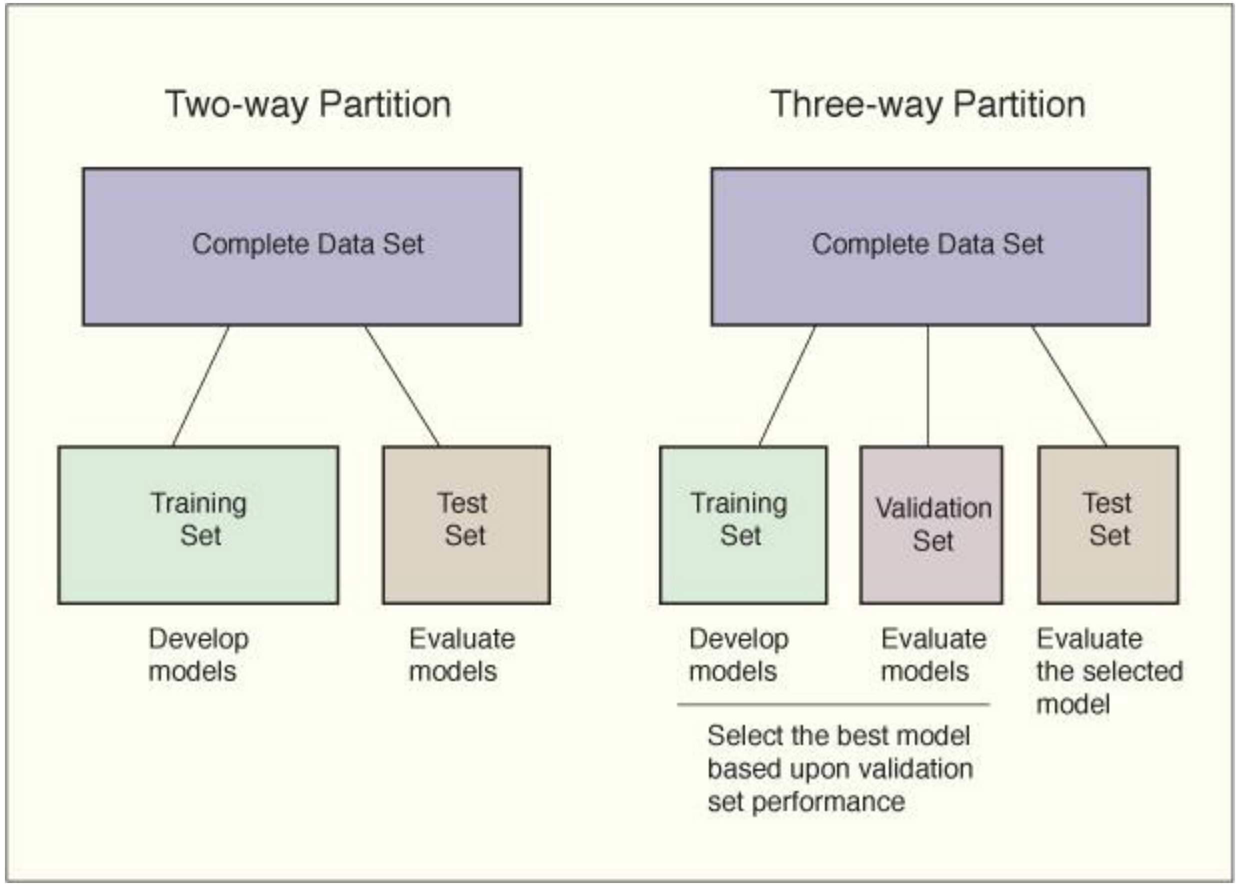
- The best predictive and fitted model would be where the validation error has its global minimum.



Model Over-fitting Due to Degrees of Freedom

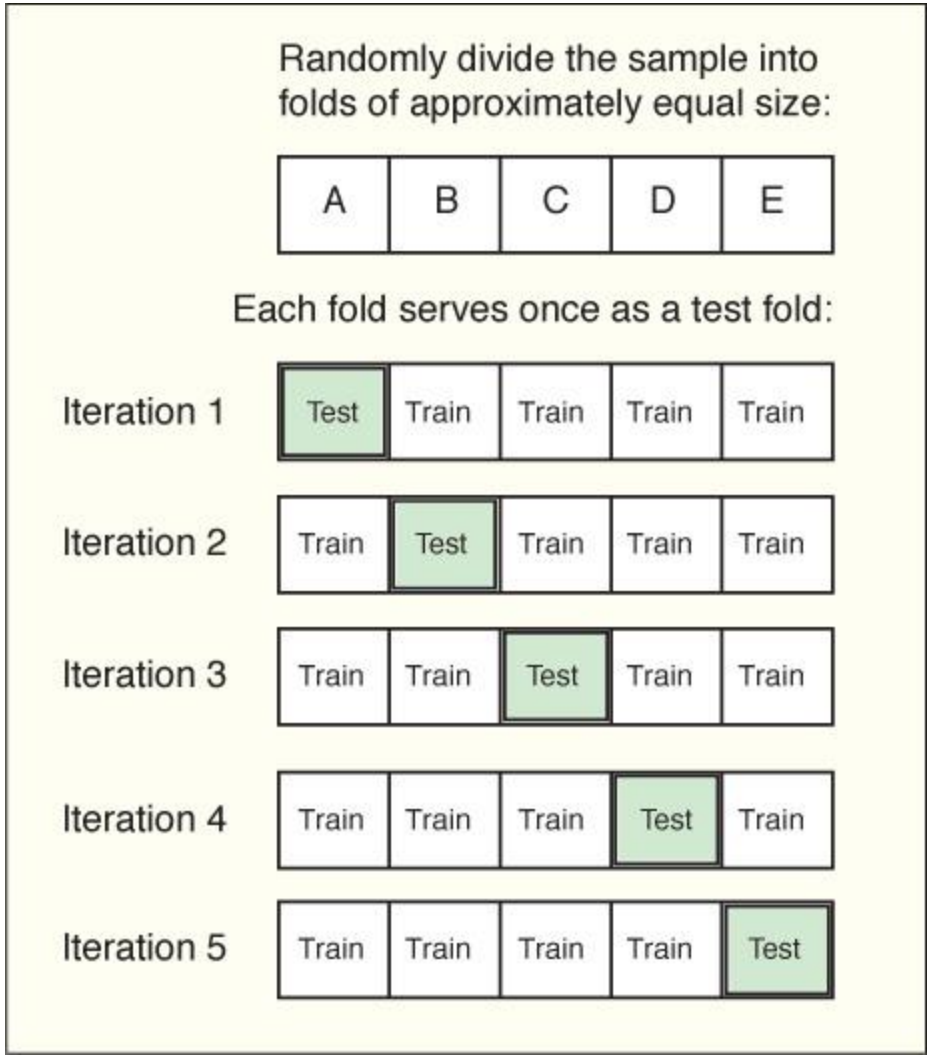


Training-and-Test Regimen for Model Evaluation



Multi-Fold Cross-Validation

A 5-fold cross validation



Leave-One-Out

- Leave-one-out cross-validation is simply n-fold cross-validation, where n = number of instances in the dataset.
- Each instance in turn is left out, and the model is trained on all remaining instances.

- Advantages:
  - Greatest possible amount of data is used for training.
  - The procedure is deterministic: no random sampling is involved, obtain the same result each time.
- Disadvantages:
  - Computationally expensive
  - Nonstratified sample

Performance Measures for Numeric Prediction

- Predicted values on the test instances are $p_1, p_2, \ldots, p_n$
- Actual values are $a_1, a_2, \ldots, a_n$

Mean-squared error	$\dfrac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{n}$
Root mean-squared error	$\sqrt{\dfrac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{n}}$
Mean-absolute error	$\dfrac{\lvert p_1-a_1\rvert+\ldots+\lvert p_n-a_n\rvert}{n}$
Relative-squared error*	$\dfrac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{(a_1-\bar{a})^2+\ldots+(a_n-\bar{a})^2}$
Root relative-squared error*	$\sqrt{\dfrac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{(a_1-\bar{a})^2+\ldots+(a_n-\bar{a})^2}}$
Relative-absolute error*	$\dfrac{\lvert p_1-a_1\rvert+\ldots+\lvert p_n-a_n\rvert}{\lvert a_1-\bar{a}\rvert+\ldots+\lvert a_n-\bar{a}\rvert}$
Correlation coefficient**	$\dfrac{S_{PA}}{\sqrt{S_P S_A}}$, where $S_{PA}=\dfrac{\sum_i(p_i-\bar{p})(a_i-\bar{a})}{n-1}$, $S_P=\dfrac{\sum_i(p_i-\bar{p})^2}{n-1}$, $S_A=\dfrac{\sum_i(a_i-\bar{a})^2}{n-1}$

**Here, $\bar{a}$ is the mean value over the training data.*
***Here, $\bar{a}$ is the mean value over the test data.*

Witten, Ian H., and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

Evaluating Predictive Accuracy of a Binary Classifier

- Confusion matrix of a binary classifier

		True Class	
		Positive	Negative
Predicted Class	Positive	True Positive Count (TP)	False Positive Count (FP)
	Negative	False Negative Count (FN)	True Negative Count (TN)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

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

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

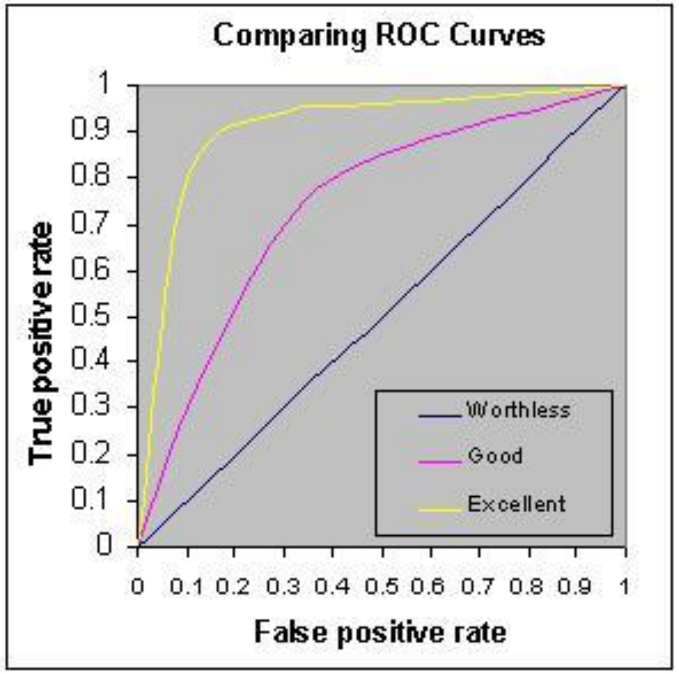
a.k.a. Sensitivity, Hit Rate, True Positive Rate

$$Specificity = \frac{TN}{TN + FP}$$

a.k.a. True Negative Rate

Evaluating Predictive Accuracy of a Binary Classifier

- ROC Curve: A plot of the true positive rate against the false positive rate for the different possible cut points of a diagnostic test.
- Good classifier has large area under curve (AUC).



General guide

- .90-1 = excellent (A)
- .80-.90 = good (B)
- .70-.80 = fair (C)
- .60-.70 = poor (D)
- .50-.60 = fail (F)

ROC = Receiver Operating Characteristic

For a more detailed explanation, watch video https://www.youtube.com/watch?v=OAl6eAyP-yo

Evaluating General Classifiers (2 or more classes)

- Cohen's Kappa coefficient is a statistic that measures inter-rater agreement for qualitative items

$$kappa = \frac{p_0 - p_e}{1 - p_e}$$

p_0: observed agreement

p_e: hypothetical probability of chance agreement

- When we have two levels of class

	Rater A (Ground Truth)			
Rater B (Classifier)		Class 1	Class 2	Total
	Class 1	p_{11}	p_{12}	$p_{1\cdot}$
	Class 2	p_{21}	p_{22}	$p_{2\cdot}$
	Total	$p_{\cdot 1}$	$p_{\cdot 2}$	

$$p_0 = p_{11} + p_{22}$$

$$p_e = p_{1\cdot}p_{\cdot 1} + p_{2\cdot}p_{\cdot 2}$$

Interpreting Kappa

Kappa Statistic	Level of Agreement
0	Equal to chance
Less than 0.20	Poor agreement
0.20 to 0.40	Fair agreement
0.40 to 0.60	Moderate agreement
0.60 to 0.80	Good agreement
0.80 to 1.00	Very good agreement

Cohen's Kappa Example

	Rater A (Ground Truth)		
Rater B (Classifier)		Class 1	Class 2
	Class 1	61	2
	Class 2	6	25

$$p_0 = p_{11} + p_{22} = \frac{61}{(61+2+6+25)} + \frac{25}{(61+2+6+25)} = \frac{61}{94} + \frac{25}{94} = \frac{86}{94} = 0.915$$

$$p_e = p_{1\cdot}p_{\cdot 1} + p_{2\cdot}p_{\cdot 2} = \frac{(61+2)}{94}\frac{(61+6)}{94} + \frac{(6+25)}{94}\frac{(2+25)}{94} = 0.572$$

$$kappa = \frac{p_0 - p_e}{1 - p_e} = \frac{0.915 - 0.572}{1 - 0.572} = 0.801$$

AGENDA

- Introduction to Predictive Analytics
- Predictive Performance Evaluation
- Using caret R Package
- Case Study: Predict Customer Churn
- Prediction and Classification Methods

Use caret Package

- caret = classification and regression training
- The caret package is a set of functions that attempt to streamline the process for creating predictive models.
- The package contains tools for:
  - data splitting
  - pre-processing
  - feature selection
  - model tuning using resampling
  - variable importance estimation
  - ……

- To learn more, visit http://topepo.github.io/caret/index.html

Simple Splitting

- A single 80/20% split of the corolla data

```r
# Read data file
df <- read.csv("ToyotaCorolla.csv")

# Use caret package
install.packages("caret", dependencies = c("Depends"))
library(caret)

# Data partition
set.seed(1234)
trainIndex <- createDataPartition(df$Price, p = .8, list = FALSE)
head(trainIndex)

train_data <- df[ trainIndex,]
test_data  <- df[-trainIndex,]
```

Simple Splitting (cont.)

- Train a linear regression model and evaluate performance

```r
# Train a linear model
m1 <- lm(Price~., data = train_data)

# Make predictions
x_test <- test_data[,2:10]
y_test <- test_data[,1]
predictions <- predict(m1, test_data)

# Summarize results
postResample(predictions, y_test)

# RMSE      Rsquared
# 1268.0328595    0.8870207
```

Advanced Modeling Training/Tuning

- Use caret::train() to tune model parameters

```
Define sets of model parameter values to evaluate
for each parameter set do
    for each resampling iteration do
        Hold–out specific samples
        [Optional] Pre–process the data
        Fit the model on the remainder
        Predict the hold–out samples
    end
    Calculate the average performance across hold–out predictions
end
Determine the optimal parameter set
Fit the final model to all the training data using the optimal parameter set
```

Tune Linear Regression

- Use 5-fold Cross-Validation

```
fitControl <- trainControl(method = "cv",number = 5)

set.seed(123)
lm_fit <- train(Price ~ ., data = df,
                trControl = fitControl,
                          method = "lm")
print(lm_fit)
# RMSE      Rsquared
# 1347.284  0.860958
```

Tune Stochastic Gradient Boosting

- also known as Gradient Boosted Machine or GBM

```
fitControl <- trainControl(method = "cv",number = 5)

set.seed(123)
gbm_fit <- train(Price ~ ., data = df,
                 trControl = fitControl,
                            method = "gbm")

print(gbm_fit)
# interaction.depth  n.trees  RMSE      Rsquared
# 1                   50      1349.728  0.8674827
# 1                  100      1224.126  0.8848723
# 1                  150      1204.684  0.8886277
# 2                   50      1193.009  0.8913350
# 2                  100      1140.768  0.8997733
# 2                  150      1134.370  0.9008928
# 3                   50      1137.894  0.9007261
# 3                  100      1110.808  0.9051474
# 3                  150      1098.506  0.9071906
```

Tune Support Vector Machine (Radial Kernel)

- Use 5-fold Cross-Validation

```
fitControl <- trainControl(method = "cv",number = 5)

set.seed(123)
svmRadial_fit <- train(Price ~ ., data = df,
                       trControl = fitControl,
                       method = "svmRadial")

print(svmRadial_fit)
#  C      RMSE      Rsquared
#  0.25   1546.399  0.8288481
#  0.50   1375.363  0.8588563
#  1.00   1297.780  0.8720767
```

Compare Multiple Models

	Linear Model with a Simple 80/20% Split	Linear Model with a 5-Fold Cross Validation	Stochastic Gradient Boosting with a 5-Fold Cross Validation	SVM (Radial Kernel) with a 5-Fold Cross Validation
RMSE	1268.033	1347.284	1098.506	1297.780
R^2	0.8870	0.8610	0.9072	0.8721

- Cross-validation can alleviate over-fitting problem
- Stochastic Gradient Boosting with a 5-fold cross validation has the best performance
  - Lowest RMSE + highest R^2

Paired t-test of Difference between Two Models

- For each metric, all pair-wise differences are computed and tested
- Null hypothesis H_0: the difference between two models is equal to zero.

```
> resamps <- resamples(list(pls = plsFit, rda = rdaFit))
> summary(resamps)
> diffs <- diff(resamps)
> summary(diffs)
```

AGENDA

- Introduction to Predictive Analytics
- Predictive Performance Evaluation
- Using caret R Package
- Case Study: Predict Customer Churn
- Prediction and Classification Methods

Customer Churn Prediction

- In telecommunication service, ***churn*** is the action that a customer's service is canceled.

- Churn analysis can help telecommunications companies to optimize their customer retention resources in order to reduce customer churn.

Image source: http://blog.soliditech.com/hs-fs/hubfs/Blog/4-Reasons-You-Shouldnt-Ignore-Customer-Churn.png

Churn Prediction Based on Customer Attributes

- Two customers and their input features.

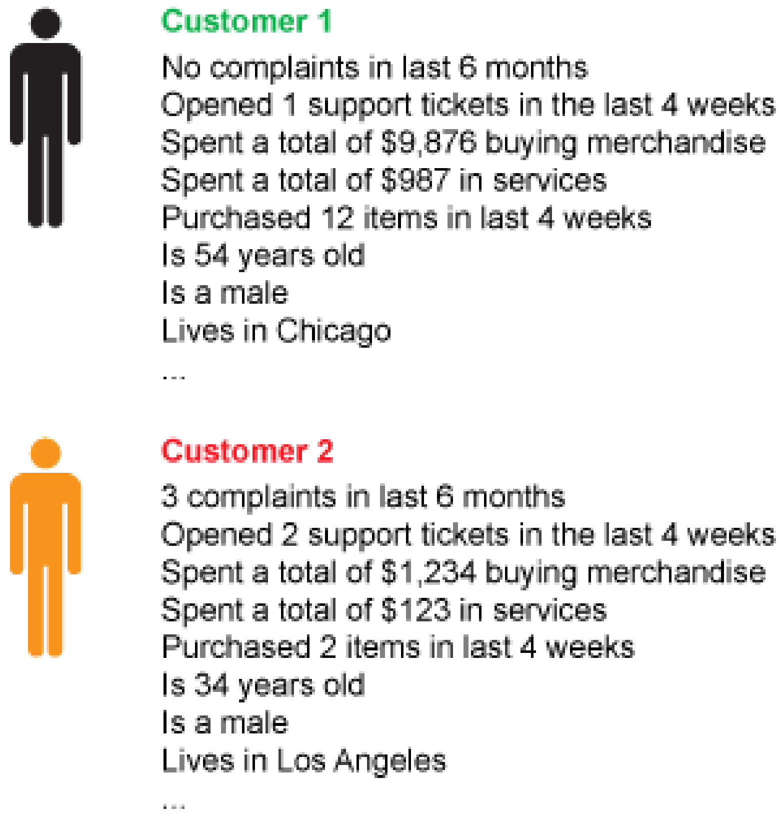
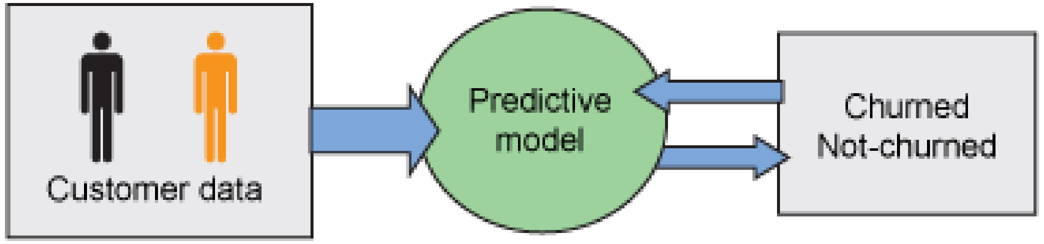


Image source: http://www.ibm.com/developerworks/library/ba-predictive-analytics1/

Basic Process of Churn Prediction

- Train a predictive model that can accurately distinguish between customers who have churned and customers who are still in service.



- Use the predictive model to monitor all existing customer activity. Use the predicted churn risk to guide business operations (such as discount promotion).

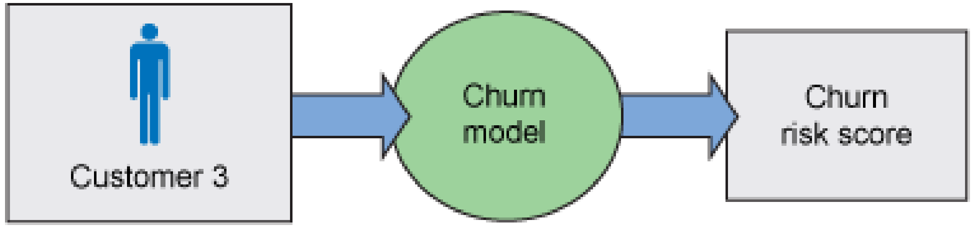


Image source: http://www.ibm.com/developerworks/library/ba-predictive-analytics1/

Dataset

- Use the Telco Customer Churn Dataset
- 7043 observations
- 21 variables:
  - ❑ CustomerID
  - ❑ Gender
  - ❑ SeniorCitizen
  - ❑ Partner
  - ❑ Dependents
  - ❑ Tenure
  - ❑ PhoneService
  - ❑ MultipleLines
  - ❑ InternetService
  - ❑ OnlineSecurity
  - ❑ OnlineBackup
  - ❑ DeviceProtection
  - ❑ TechSupport
  - ❑ StreamingTV
  - ❑ StreamingMovies
  - ❑ Contract
  - ❑ PaperlessBilling
  - ❑ PaymentMethod
  - ❑ MonthlyCharges
  - ❑ TotalCharges
  - ❑ Churn

Data source: https://community.watsonanalytics.com/wp-content/uploads/2015/03/WA_Fn-UseC_-Telco-Customer-Churn.csv

Customer Churn Prediction – Model Selection

- In this example, we'll explore three different methods to predict customer churn
  - Logistic Regression
  - Support Vector Machine (SVM)
  - Gradient Boosted Machine (GBM)

R Code

- Refer to R Markdown Report

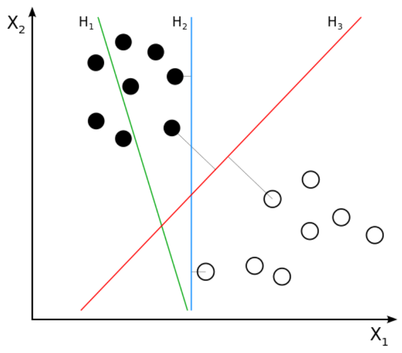
PredictCustomerChurn.pdf

AGENDA

- Introduction to Predictive Analytics
- Predictive Performance Evaluation
- Using caret R Package
- Case Study: Predict Customer Churn
- Prediction and Classification Methods

Predictive Analytical Methods

- Time series
  - Statistical techniques that use historical demand data to predict future demand
  - Only require historical data on the variable to predict itself
- Regression methods
  - Attempt to develop a mathematical relationship between demand and factors that cause its behavior
  - Require historical data of both DV and IVs
- Advanced data mining approaches
  - Decision trees
  - Artificial neural networks (deep learning)
  - Support vector machines
  - Bayesian classifiers
  - ……



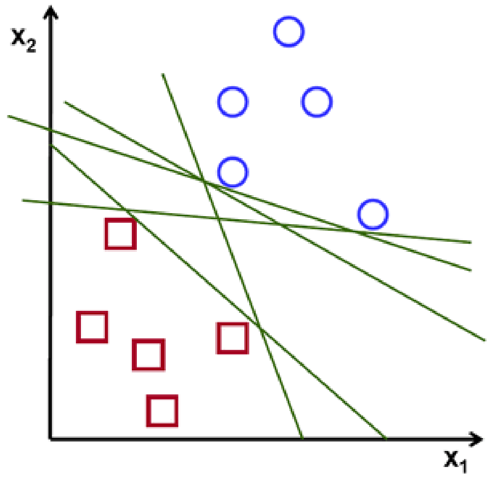
Regression

- Choose the appropriate regression model based on response variable

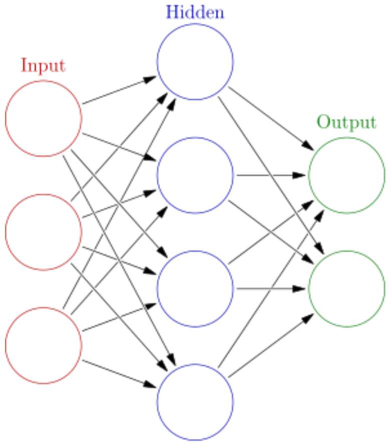
Response Variable	Regression Model
Ratio data (e.g., price)	Linear regression
Binary data (e.g., yes/no, 1/0, die/live)	Logistic, probit
Counts (e.g., number of visits, number of patents granted)	Poisson, negative binomial
Duration (e.g., survival time after heart attack)	Survival analysis
Discrete choice (>=3 categories)	Multinomial logit, multinomial probit
Cornered, censored (value of response variable is limited in a range, e.g., from 0 to 10)	Truncated regression, interval regression, Tobit etc.

Machine Learning

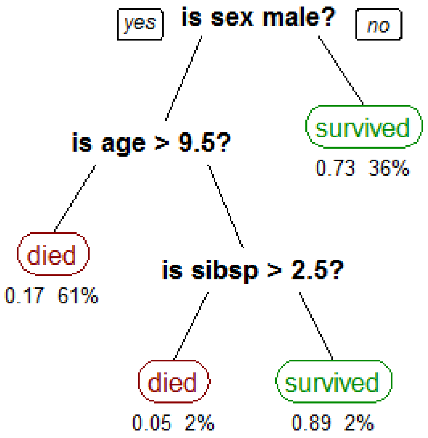
- Machine learning methods often perform better than traditional regression methods, but explaining why they work is usually difficult.
- Many machine learning methods are *black box* models.



Support Vector Machine



Neural Network



Decision Tree

Supervised and Unsupervised Learning

- Supervised Learning
  - Supervised learning algorithms are used for prediction and classification.
  - We need to supervise the learning of the algorithm by using training data to train the algorithm.
  - Data is labeled.

- Unsupervised Learning
  - Unsupervised learning algorithms are used when there is no outcome variable to predict or classify.
  - Data is unlabeled.
  - There is no training-testing partition of the dataset.
  - For example, association rules, clustering.

Some Predictive Analytics Methods

- k-NN (k-Nearest-Neighbors)
- Naïve Bayes
- Neural Network
- SVM (Support Vector Machine)
- Ensembles

k-Nearest-Neighbors for Classification and Prediction

- K-NN algorithm find "similar" records in the training data, then use these "neighbors" to derive a classification or prediction for the new record.
  - Classification: Assign a class by voting among neighbors
  - Prediction: Create prediction by averaging across neighbors

Measuring Distance between Records

- The step of finding neighbors depends on distance metrics.

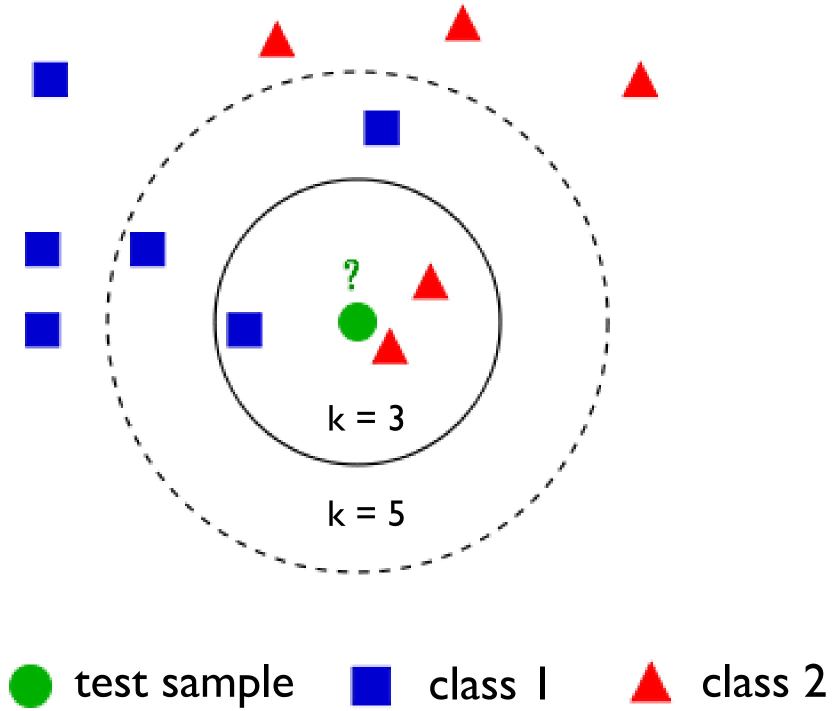
- For continuous variables, a commonly used distance metric is Euclidean distance.

$$d(p,q) = d(q,d) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

- We can use other metrics such as standardized Euclidean distance, Mahalanobis distance, Minkowski distance, Chebychev distance, Cosine distance, Hamming distance, Manhattan distance, Jaccard distance, Spearman distance etc.

K-NN Classification

- The choice of parameter k
  - Large values of k reduce the effect of noise, but make boundaries between classes less distinct.



If k = 3, the test sample is classified as class 2;

If k = 5, class 1 is assigned.

k-NN Prediction (Regression)

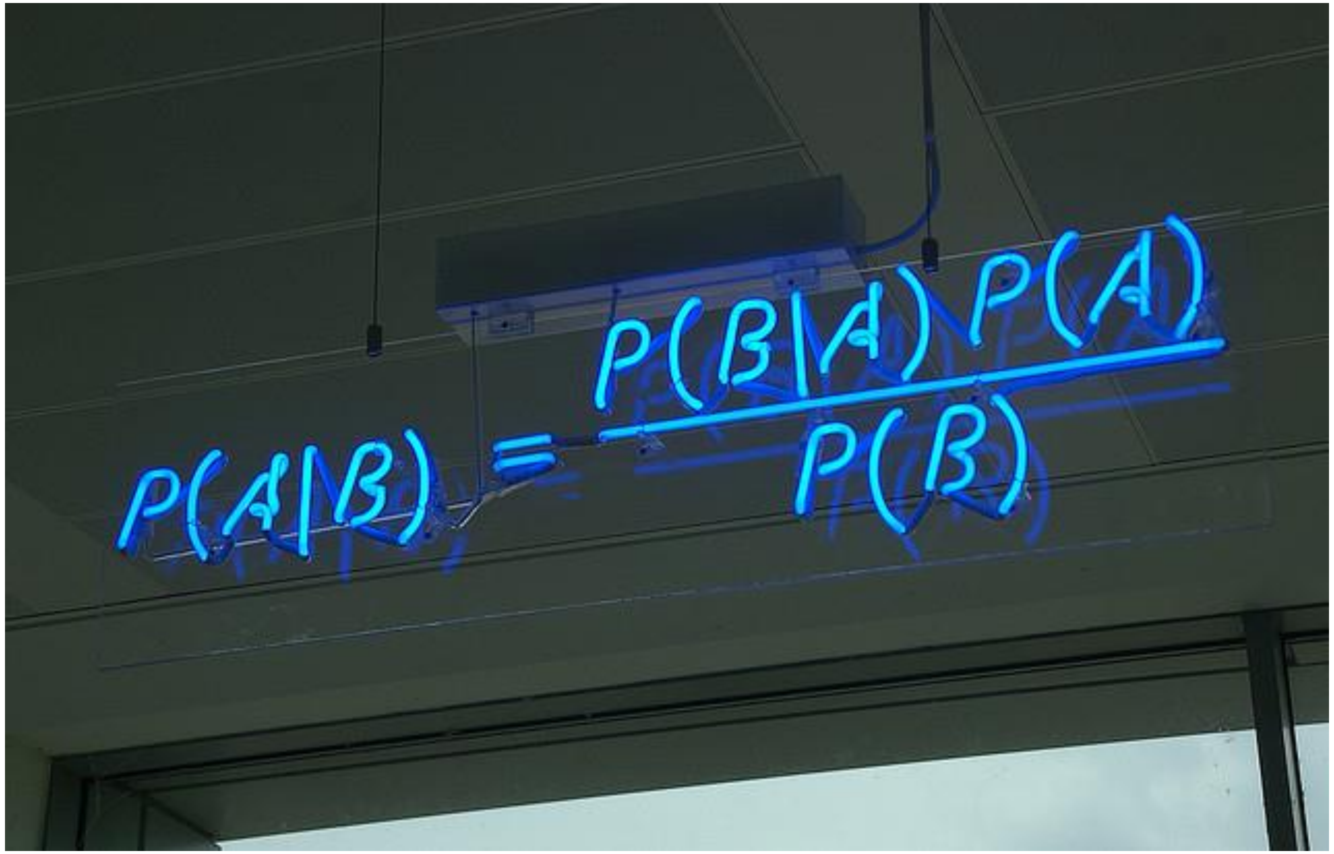
- A Simple Modification to the voting mechanism for classification:
  - Step 1. Find neighbors by calculating distances;
  - Step 2. Take the average response value of the k-nearest-neighbors as the prediction for the focal record.

Summary of k-NN

- Advantages
  - A nonparametric method without assumption about the relationship between X and Y;
  - Accuracy is good with a large enough training data;
  - It has minimal configuration (the only parameter is *k*, the number of neighbors)

- Disadvantages
  - Need a long time to compute distance with a very large dataset;
  - The number of records required in the training data to qualify as large enough increases exponentially with the number of predictors p;
  - k-NN is a "lazy learner": the time-consuming computation is deferred to the time of prediction.
    - It's not applicable for real-time prediction with large dataset.

Bayes' Theorem



Bayes' Theorem

e = event, D = data

$$P(e|D) = \frac{P(D|e)P(e)}{P(D)} \propto P(D|e)P(e)$$

- $P(e)$: prior probability, what we know about e without any information
- $P(D|e)$: conditional probability or likelihood, what we assume to be true
- $P(e|D)$: posterior probability of event e given information D, what we want to know

Posterior ∝ Likelihood * Prior

Complete/Exact Bayes Classifier

- Define Classification Problem
  - For a response with m classes $C_1, C_2, \dots, C_m$, and the predictor variables $x_1, x_2, \dots, x_p$, we want to know:

$$P(C_i | x_1, x_2, \dots, x_p)$$

- Complete Bayes Classifier
  - Calculate conditional probability:

$$P(C_i | x_1, x_2, \dots, x_p) = \frac{P(x_1, x_2, \dots, x_p | C_i) P(C_i)}{P(x_1, x_2, \dots, x_p | C_1) P(C_1) + \cdots + P(x_1, x_2, \dots, x_p | C_m) P(C_m)}$$

  - Assign class based on the conditional probability:
    - Assign to the most probable class
    - Assign to the class with probability >= cutoff

Naïve Bayes Classifier

- Make "naïve" assumption of conditional independence among predictors

$$P(x_1,x_2,\dots,x_p|C_i)=P(x_1|C_i)P(x_2|C_i)\cdots P(x_p|C_i)=\prod_{j=1}^{p} P(x_j|C_i)$$

- Naive Bayes Classifier
  - Calculate conditional probability:

$$P(C_i|x_1,x_2,\dots,x_p)=\frac{P(x_1,x_2,\dots,x_p|C_i)P(C_i)}{P(x_1,x_2,\dots,x_p|C_1)P(C_1)+\cdots+P(x_1,x_2,\dots,x_p|C_m)P(C_m)}$$
$$=\frac{P(C_i)\prod_{j=1}^{p}P(x_j|C_i)}{\sum_{i=1}^{m}\left[P(C_i)\prod_{j=1}^{p}P(x_j|C_i)\right]}$$

  - Assign class based on the conditional probability:
    - Assign to the most probably class
    - Assign to the class with probability >= cutoff

Information Needed for Naïve Bayes

- To use the naïve Bayes classifier, we only need the following data:
  - $P(C_i)$: the prior probability for class C_i
  - $P\left(x_j \middle| C_i\right)$: the conditional probability (or likelihood) of feature x_j given class C_i

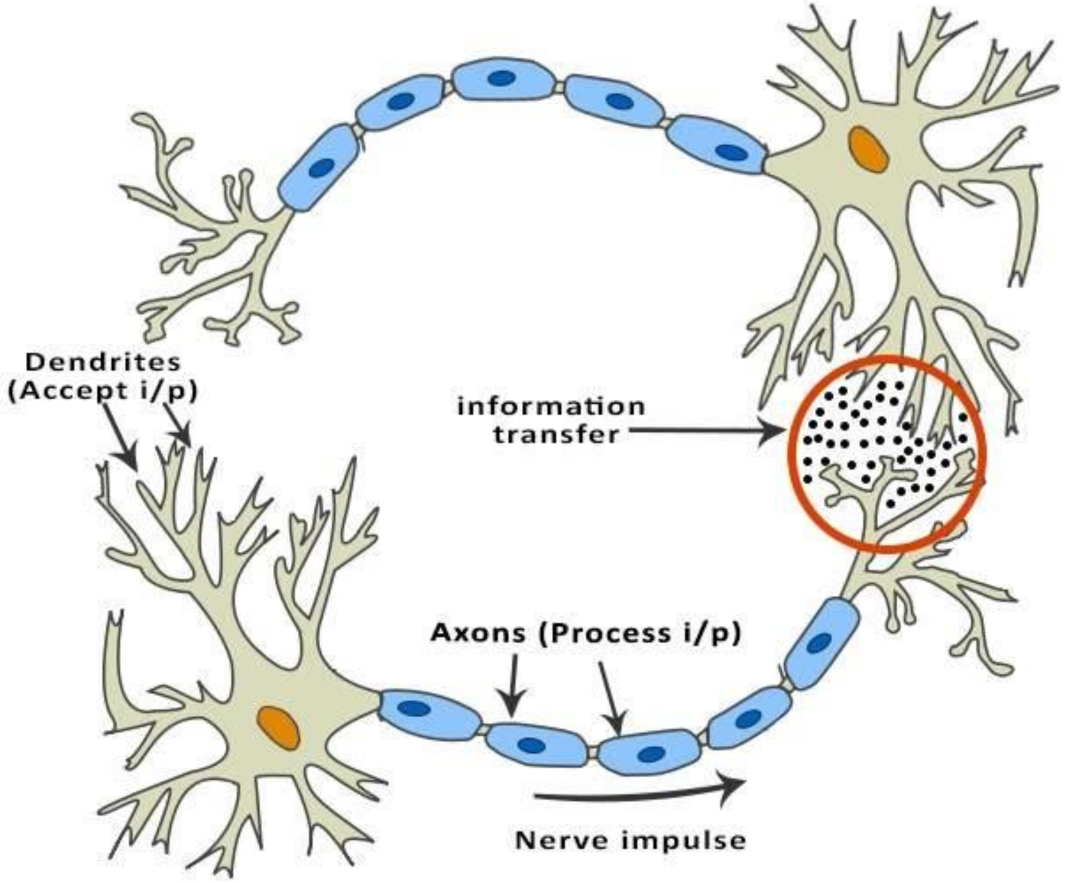
Summary of Naïve Bayes Classifier

- Advantages
  - It is "naïve" and simple: Naïve Bayes classifier is simple to implement and has good computational efficiency.
  - It has good performance when the input variables are categorical. Naïve Bayes classifier can directly handle categorical variables.
  - It performs well even when the conditional independence assumption is violated.

- Disadvantages
  - The probability estimate of class (propensity) is biased. Naïve Bayes classifier is rarely used in credit scoring.
  - Cannot model structural patterns.

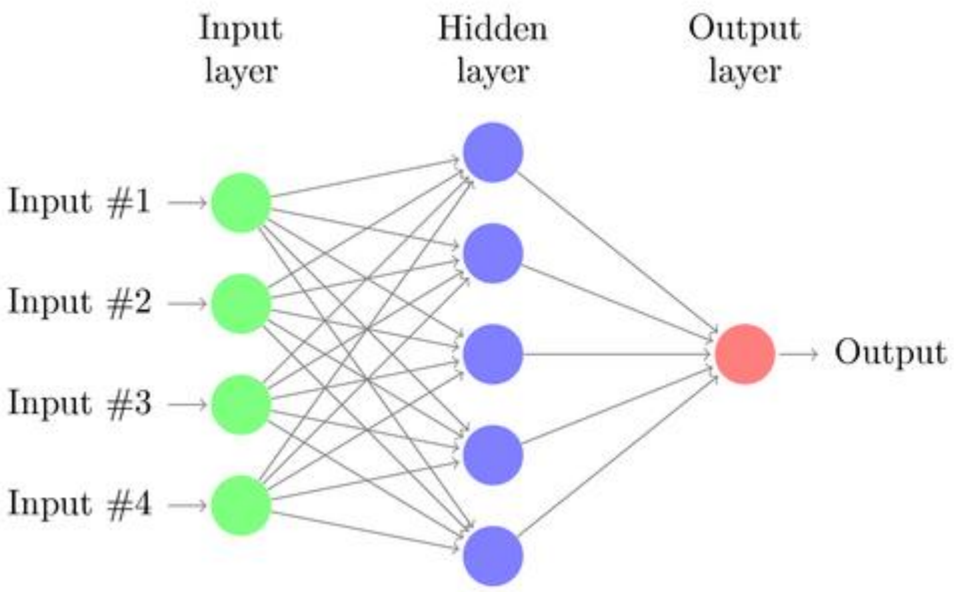
How Human Brain Works?

- The human brain is comprised of billions o nerve cells, neurons, that are interconnected by axons.
- Dendrites accept stimuli from external environment (inputs)
- The inputs travels through the neural network though neurons and axons.
- Neurons learn from experience to handle information.



ANN Models the Human Brain

- A typical artificial neural network
  - An input layer accepts input data
  - An output layer provides output
  - Hidden layers connect the input and output layers



ANN tries to learn the underlying relationship between input (predictors) and output (responses).

Fitting a Neural Net to Data

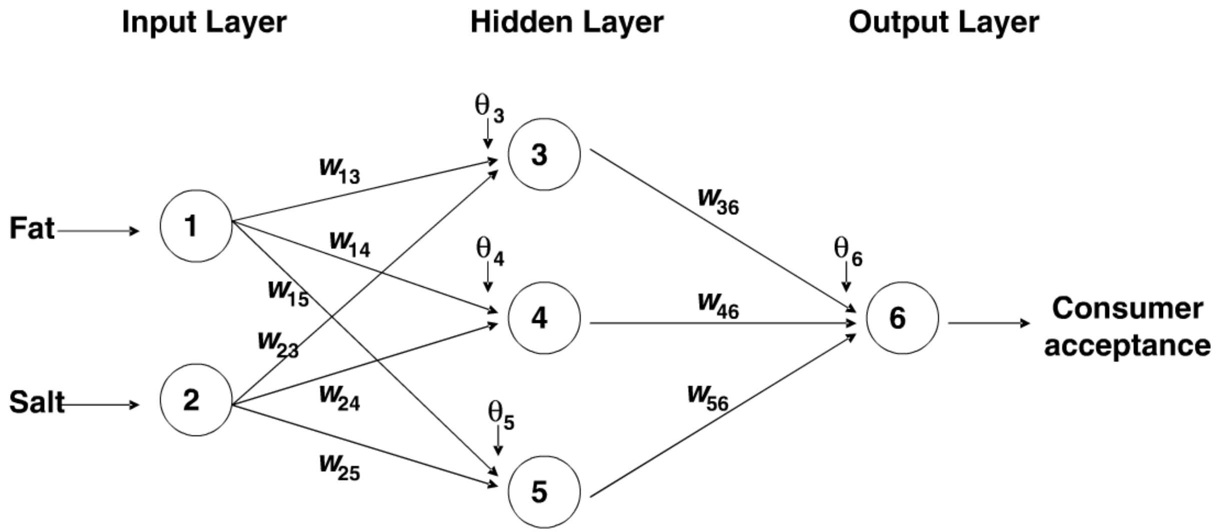
- Data: testing scores for 6 consumers and two predictors

Obs.	Fat Score	Salt Score	Acceptance
1	0.2	0.9	1
2	0.1	0.1	0
3	0.2	0.4	0
4	0.2	0.5	0
5	0.4	0.5	1
6	0.3	0.8	1

- The ANN model to fit

$W_{i,j}$: *weights*

θ_j: *node bias*



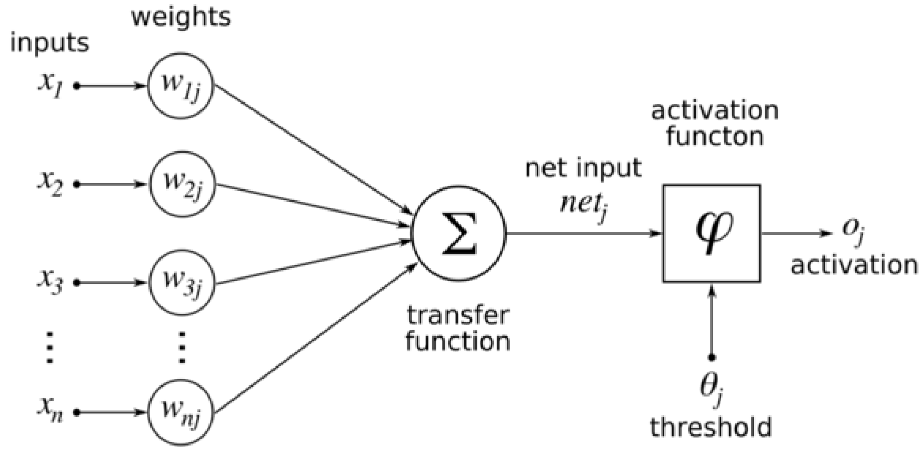
Relation to Linear and Logistic Regression

- Consider an ANN with a single output and no hidden layers

  - A numerical output y, identity activation function g(s)=s

$$\hat{y} = \theta + \sum_{i=1}^{n} w_i x_i$$

ANN ≈ Multiple Linear Regression



  - A binary output y, activation function is in logistic form

$$p(y=1) = \frac{1}{1 + exp\left[-\left(\theta + \sum_{i=1}^{n} w_i x_i\right)\right]}$$

ANN ≈ Logistic Regression

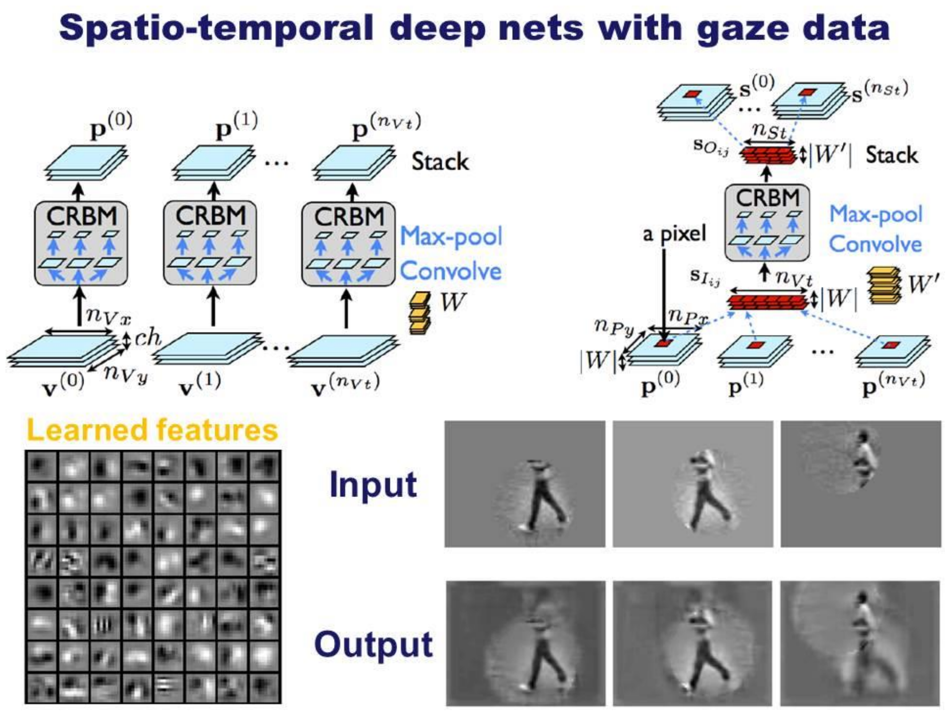
Summary of Neural Nets for Classification and Prediction

- Neural networks are black boxes: cannot interpret the underlying relationships.
- Otherwise, neural networks generally have good predictive performance.
- Requirement on sufficient data for training the model.
- Relatively heavy on computation time.

Deep Learning Networks (DLN)

- With improvements in computing power, deep learning networks are popularly used to deal with big data and extreme complexity of the networks.

- Deep learning networks refer to neural nets with many hidden layers used to self-learn features from the complex data.



Source: https://www.cs.ox.ac.uk/projects/DeepLearn/

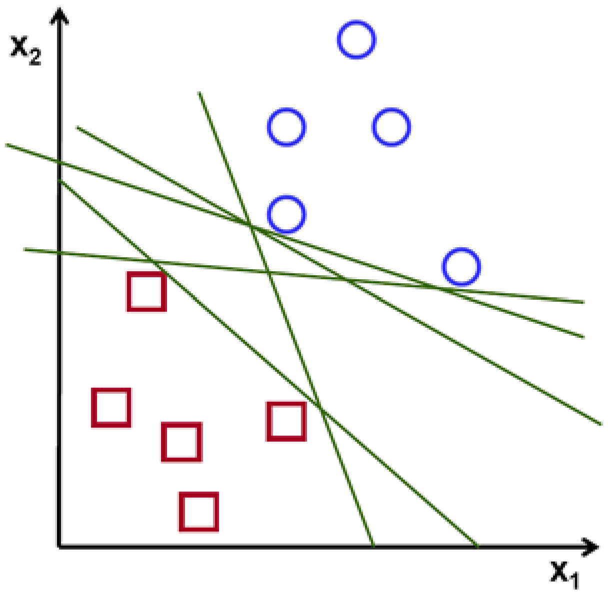
R Code

- Refer to R Markdown Report

PredictCorollaPrice_ANN.pdf

Support Vector Machine (SVM)

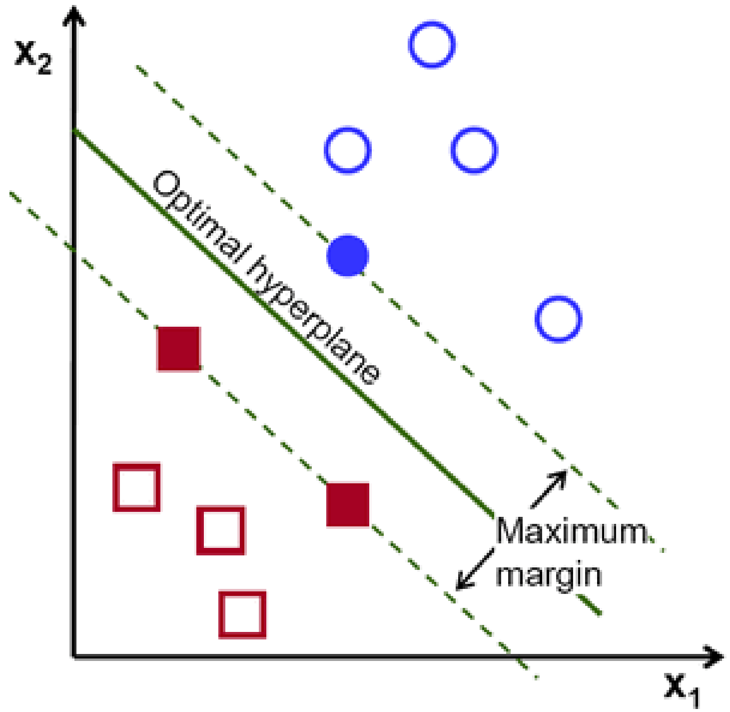
- A Support Vector Machine (SVM) is a discriminative classifier that uses an optimal hyperplane to separate different classes.
- A classification example in a two dimensional space
  - What is the optimal line to separate the points in two classes?



Source: http://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html

SVM (cont.)

- SVM algorithm tries to find the hyperplane that gives the largest minimum distance (margin) to the training examples.
- The optimal separating hyperplane *maximizes* the margin of the training data.



To learn more, read: http://www.cs.columbia.edu/~kathy/cs4701/documents/jason_svm_tutorial.pdf

Ensembles

- An ensemble combines multiple supervised models into a "supermodel".
- Three ways of creating ensembles
  - Simple average: to average the prediction, or select by voting for classification;
  - Bagging (a.k.a. boostrap aggregating): to average across multiple random samples;
  - Boosting: to improve areas in the data where there are large prediction errors.

> "An early lesson of the competition was the value of combining sets of predictions from multiple models or algorithms. If two prediction sets achieved similar RMSEs, it was quicker and more effective to simply average the two sets than to try to develop a new model that incorporated the best of each method. Even if the RMSE for one set was much worse than the other, there was almost certainly a linear combination that improved on the better set."

Bell et al. "All Together Now: A Perspective on the NETFLIX PRIZE"

https://amba-bigdata.wikispaces.com/file/view/Netflix_general.pdf

An Ensemble: Gradient Boosting Machine (GBM)

- GBM is one of boosting algorithms that convert weak learners (typically decision trees) to strong learners

Algorithm 1 Friedman's Gradient Boost algorithm

Inputs:

- input data $(x, y)_{i=1}^{N}$
- number of iterations M
- choice of the loss-function $\Psi(y, f)$
- choice of the base-learner model $h(x, \theta)$

Algorithm:

1: initialize $\widehat{f}_0$ with a constant
2: **for** $t = 1$ to M **do**
3: compute the negative gradient $g_t(x)$
4: fit a new base-learner function $h(x, \theta_t)$
5: find the best gradient descent step-size ρ_t:

$$\rho_t = \arg\min_\rho \sum_{i=1}^{N} \Psi\left[y_i, \widehat{f}_{t-1}(x_i) + \rho h(x_i, \theta_t)\right]$$

6: update the function estimate:

$$\widehat{f}_t \leftarrow \widehat{f}_{t-1} + \rho_t h(x, \theta_t)$$

7: **end for**

To learn more, visit https://en.wikipedia.org/wiki/Gradient_boosting

Q & A